Dynamic Environment Responsive Online Meta-Learning with Fairness Awareness

CHEN ZHAO, Baylor University, Waco, Texas, USA
FENG MI, The University of Texas at Dallas, Richardson, Texas, USA
XINTAO WU, University of Arkansas, Fayetteville, Arkansas, USA
KAI JIANG, The University of Texas at Dallas, Richardson, Texas, USA
LATIFUR KHAN, The University of Texas at Dallas, Richardson, Texas, USA
FENG CHEN, The University of Texas at Dallas, Richardson, Texas, USA

The fairness-aware online learning framework has emerged as a potent tool within the context of continuous lifelong learning. In this scenario, the learner's objective is to progressively acquire new tasks as they arrive over time, while also guaranteeing statistical parity among various protected sub-populations, such as race and gender when it comes to the newly introduced tasks. A significant limitation of current approaches lies in their heavy reliance on the i.i.d (independent and identically distributed) assumption concerning data, leading to a static regret analysis of the framework. Nevertheless, it's crucial to note that achieving low static regret does not necessarily translate to strong performance in dynamic environments characterized by tasks sampled from diverse distributions. In this article, to tackle the fairness-aware online learning challenge in evolving settings, we introduce a unique regret measure, FairSAR, by incorporating long-term fairness constraints into a strongly adapted loss regret framework. Moreover, to determine an optimal model parameter at each time step, we introduce an innovative adaptive fairness-aware online meta-learning algorithm, referred to as FairSAOML. This algorithm possesses the ability to adjust to dynamic environments by effectively managing bias control and model accuracy. The problem is framed as a bi-level convex-concave optimization, considering both the model's primal and dual parameters, which pertain to its accuracy and fairness attributes, respectively. Theoretical analysis yields sub-linear upper bounds for both loss regret and the cumulative violation of fairness constraints. Our experimental evaluation of various real-world datasets in dynamic environments demonstrates that our proposed FairSAOML algorithm consistently outperforms alternative approaches rooted in the most advanced prior online learning methods.

CCS Concepts: • **Computing methodologies** → **Artificial intelligence**; **Machine learning**; • **Applied computing** → *Law, social and behavioral sciences*; • **Social and professional topics** → User characteristics;

Additional Key Words and Phrases: Fairness, online meta-learning, changing environments, adaption

1 INTRODUCTION

In the real world, data that includes biases are often collected incrementally over time, and the underlying distribution assumptions can undergo significant changes at critical junctures. A case in point is a recent report by the *New York Times* [17], which highlights that systematic algorithms exhibited increased discriminatory tendencies towards African Americans in the context of bank loans during the COVID-19 pandemic compared to the pre-pandemic era. These algorithms are constructed from a series of sequentially gathered data streams, where decision-making exhibits bias towards the protected racial population at each step. This situation underscores two key issues: (1) Online algorithms typically neglect the crucial aspect of fairness in learning, where fairness is defined as the equality of predictive performance across different sub-populations, ensuring that a model's predictions remain statistically independent of protected characteristics (e.g., race). (2) Machine learning models heavily rely on the i.i.d assumption, which becomes untenable when the environment undergoes changes, as exemplified by shifts occurring before and after the pandemic.

To effectively manage bias over time, particularly in the context of ensuring fairness across various protected sub-populations, fairness-aware online algorithms are designed to address supervised learning problems where fairness is a prominent concern. These algorithms aim to sequentially train predictive models that remain unbiased. In particular, the objective of these algorithms is twofold: first, to ensure that the static loss regret, which measures the cumulative loss of the learner against the best-fixed action in hindsight, and second, to limit the violation of various fairness principles, both exhibit sub-linear growth in the total number of time steps [28]. It's worth noting that while these approaches achieve cutting-edge theoretical guarantees, it's important to recognize that the metric of static regret holds significance primarily in stable or stationary environments. Low static regret, however, doesn't necessarily translate to excellent performance in changing environments because time-invariant benchmarks may perform poorly under such circumstances [23].

To overcome the challenge posed by changing environments in online learning, two distinct notions of regret have garnered attention: **strongly adaptive regret (SAR)** [3] and dynamic regret [30]. These concepts offer differing perspectives on handling changes over time. Dynamic regret takes a global approach, addressing changes in environments by comparing the cumulative loss of the learner against a sequence of comparators. Importantly, it allows these comparators to evolve over time, reflecting the dynamic nature of the learning process. Conversely, SAR adopts a more localized viewpoint, giving greater consideration to short time intervals. This type of regret can be seen as the maximum regret statistic across all intervals [3]. While some recent works [12, 23, 24] have made strides in achieving sub-linear loss regret in online learning within changing environments, they often overlook the crucial aspect of learning with fairness. This neglect of fairness, which is a fundamental characteristic of human intelligence, remains a significant limitation in these approaches.

In this article, we present a new challenge, namely fairness-aware online meta-learning in changing environments. In this scenario, a series of data batches or tasks are collected sequentially over time, with the environments associated with these tasks potentially undergoing variations. Our primary objectives in this research are twofold: Firstly, we aim at extending the applicability of predictive learning accuracy and model fairness to novel and evolving environments. Secondly,

we endeavor to minimize both loss regret and the cumulative violation of fairness constraints, ensuring that they exhibit sublinear growth over time.

To achieve these goals, we introduce a novel online learning algorithm named **fair strongly adaptive online meta-learner (FairSAOML)**. This algorithm updates model parameters through a two-level approach: online fair interval-level learning and meta-level learning. These two levels of problems interact with two sets of parameters: primal parameters $\boldsymbol{\theta}$, which pertain to model accuracy, and dual parameters $\boldsymbol{\lambda}$, which govern fairness considerations. To provide more details, we draw inspiration from the concept of *learning with expert advice* [12], and we carefully design three alternative sets of intervals. At each time step $t \in [T]$, a subset of intervals is chosen to activate several experts, with each active expert running an interval-specific algorithm. An expert takes a meta-solution pair $(\boldsymbol{\theta}_{t-1}, \boldsymbol{\lambda}_{t-1})$ from the previous time as input and generates an interval-level solution $(\boldsymbol{\theta}_{t,I}, \boldsymbol{\lambda}_{t,I})$ for the specific interval I. A meta-algorithm combines the weighted contributions of all experts to form a solution pair $(\boldsymbol{\theta}_t, \boldsymbol{\lambda}_t)$ at time t, which is then utilized to make predictions for the subsequent time step. This approach allows us to address the challenges of fairness-aware online meta-learning in changing environments effectively.

The main contributions of this article are summarized:

— In this article, we propose a novel framework addressing the problem of fairness-aware online meta-learning in changing environments. We start with the introduction of a novel adaptive fairness-aware regret FairSAR. A novel algorithm FairSAOML is further proposed to find a good decision sequentially. At each time, the problem is formulated as a constrained bi-level convex-concave optimization with respect to a primal-dual parameter pair.

— Based on varying assumptions and motivations, we introduce three distinct sets of intervals, leading to the creation of three different versions of our proposed FairSAOML algorithm.

— Theoretically grounded analysis justifies the efficiency and effectiveness of all variants of FairSAOML by demonstrating tighter bounds $O((\tau \log T)^{1/2})$ for the loss regret and $O((\tau T \log T)^{1/4})$ for violation of fairness constraints.

— We validate the performance of our approach with state-of-the-art techniques on real-world datasets. Our results demonstrate that FairSAOML can effectively adapt both accuracy and fairness in changing environments, and it shows substantial improvements over the best prior works.

This article is organized as follows. In Section 2, some related works are introduced. Section 3 provides notations and some backgrounds of this article. In Section 4, we detail the proposed methodology. In Section 5, we discuss the theoretically grounded analysis for the learning approach. Empirical settings and results on real-world benchmarks compared with cutting-edge techniques are given in Sections 6 and 7. Finally, this article is concluded in Section 8.

2 RELATED WORK

Changing environments in online learning. Since the pioneering work [30] in online learning, numerous subsequent researches [9, 21] have been developed under the assumption of a stationary environment with static regret. Low static regret, however, cannot imply a good performance in a changing environment due to time-invariant comparators. To address this limitation, two regret metrics, dynamic regret [30] and **adaptive regret (AR)** [10], is devised to measure the learner's performance in changing environments. To bind the general dynamic regret, the path-length of comparators [24, 30] is introduced and further developed. Unlike dynamic regret, AR handles changing environments from a local perspective by focusing on comparators in short intervals. To reduce the time complexity of AR-based online algorithms, geometric covering intervals [3, 12, 23] and data streaming techniques [7] are developed. Although existing methods achieve

state-of-the-art performance, a major drawback is that they immerse in minimizing objective functions but ignore the model fairness of prediction.

Fairness-aware online learning problems assume individuals arrive one at a time and the goal of such algorithms is to train predictive models free from biases. From the perspective of optimization, group fairness notions are normally considered as constraints added to learning objectives. However, when the constraints are complex, the computational burden of the projection onto constraints may be too high. Several closely related works, including FairFML [28], FairGLC [22], and FairAOGD [11], aim at improving the theoretic guarantees by relaxing the output through a simpler closed-form projection. However, these methods are not ideal for continual lifelong learning with changing task distributions, as they assume that all samples come from the same data distribution.

Online meta-learning addresses the issue of learning with fast adaptation, where a meta-learner learns knowledge transfer from history tasks onto new coming ones. FTML [6] can be considered as an application of MAML [5] in the setting of online learning. FairFML [28] extends FTML by controlling bias in an online working paradigm with task-specific adaptation. Unfortunately, none of such techniques are devised to adapt to changing environments.

Although a recent work [29] tackles the problem of fairness-aware online learning for changing environments, it heavily depends on the assumption that the number of times is known in advance and unchanged. The number of learning processes is hence fixed. Besides, due to the setting of intervals in this work, the learning efficiency at the beginning times is low.

In this article, to bridge the above-mentioned areas, we study the problem of fairness-aware online meta-learning to deal with changing task environments. In particular, at each time, model parameters are determined by the proposed novel algorithm FairSAOML. This algorithm refers to ideas of dynamic programming and expert tracking techniques. Inspired by fairness-aware online learning and meta-learning, a bi-level adaptation strategy is used to accommodate changing environments and learn models with accuracy and fairness.

3 PRELIMINARIES

3.1 Notations

An index set of a sequence of tasks is defined as $[T] = \{1, 2, \ldots, T\}$ and $[t, T] = \{t, t + 1, \ldots, T\}$. Vectors are denoted by lowercase boldface letters. Scalars are denoted by lowercase italic letters. Some important notations are listed in Table 1.

3.2 Constraints for Group Fairness

In general, group fairness criteria used for evaluating and designing machine learning models focus on the relationships between the protected attribute and the system output [19, 26, 27]. The problem of group unfairness prevention can be seen as a constrained optimization problem. For simplicity, we consider one binary protected attribute (e.g., gender) in this work. However, our ideas can be easily extended to many protected attributes with multiple levels.

Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be the data space, where $\mathcal{X} = \mathcal{E} \cup \mathcal{S}$. Here $\mathcal{E} \subset \mathbb{R}^d$ is an input space, $\mathcal{S} = \{-1, 1\}$ is a protected space, and $\mathcal{Y} = \{-1, 1\}$ is an output space for binary classification. Given a task (batch) of samples $\{\mathbf{e}_i, y_i, s_i\}_{i=1}^n \in (\mathcal{E} \times \mathcal{Y} \times \mathcal{S})$ where n is the number of datapoints, a fine-grained measurement to ensure fairness in class label prediction is to design fair classifiers by controlling the notions of fairness between protected subgroups, such as **demographic parity** (**DP**) and equality of opportunity [14, 20].

Definition 1 (Notions of Fairness [14, 20]). A classifier $h : \Theta \times \mathbb{R}^d \to \mathbb{R}$ is fair when its predictions are independent of the protected attribute $\mathbf{s} = \{s_i\}_{i=1}^n$. To get rid of the indicator function and relax the exact values, a linear approximated form of the difference between protected subgroups

Table 1. Important Notations and Corresponding Descriptions

Notations	Descriptions
T	Total number of learning tasks
t	Indices of tasks
τ	Length of time intervals in general
$\mathcal{D}_t^S, \mathcal{D}_t^V, \mathcal{D}_t^Q$	Support/Validation/Query set of data $\mathcal{D}_t$
θ_t, λ_t	Meta-level primal/dual parameters at round t
$\theta_{t,I}, \lambda_{t,I}$	Interval-level primal/dual parameters for an expert E_I at round t
$f_t(\cdot)$	Loss function at round t
$g_i(\cdot)$	Fairness function
m	Total number of fairness notions
i	Indices of fairness notions
$\mathcal{G}(\cdot)$	Base learner
$\mathcal{U}$	Expert set
$\mathcal{A}_t, \mathcal{S}_t$	Active/Sleeping expert set at round t
$\mathcal{I}$	AGC interval set
C_t	Target set of intervals at round t
$\mathcal{B}$	Relaxed primal domain
$\prod_{\mathcal{B}}$	Projection operation onto domain $\mathcal{B}$
η_1, η_2	Learning rates
$p_{t,I}$	Expert weight of E_I at round t
δ	Augmented constant

is defined [14],

$$g(\theta) = \left| \mathbb{E}_{(e,y,s)\in\mathcal{Z}} \left[\frac{1}{\hat{p}_1(1-\hat{p}_1)} \left(\frac{s+1}{2} - \hat{p}_1 \right) h(\theta, e) \right] \right| - \epsilon, \qquad (1)$$

where $|\cdot|$ is the absolute function and $\epsilon > 0$ is the fairness relaxation determined by empirical analysis. $\hat{p}_1$ is an empirical estimate of pr_1. pr_1 is the proportion of samples in group $s = 1$ and correspondingly $1 - pr_1$ is the proportion of samples in group $s = -1$.

Notice that, in Definition 1, when $\hat{p}_1 = \mathbb{P}_{(e,y,s)\in\mathcal{Z}}(s = 1)$, the fairness notion $g(\theta)$ is defined as the **difference of demographic parity (DDP)**. Similarly, when $\hat{p}_1 = \mathbb{P}_{(e,y,s)\in\mathcal{Z}}(y = 1, s = 1)$, $g(\theta)$ is defined as the **difference of equality of opportunity (DEO)** [14]. Therefore, parameters θ in the domain of a task is feasible if it satisfies the fairness constraint $g(\theta) \leq 0$.

3.3 Fairness-Aware Online Learning

The protocol of fairness-aware online convex optimization can be viewed as a repeated game between a learner and an adversary, where the learner is faced with tasks $\{\mathcal{D}_t\}_{t=1}^T$ one after another. At each round $t \in [T]$,

- **Step 1**: The learner selects a model parameter θ_t in the fair domain Θ.
- **Step 2**: The adversary reveals a loss function $f_t : \Theta \times \mathbb{R}^d \to \mathbb{R}$ and m fairness functions $g_i : \Theta \times \mathbb{R}^d \to \mathbb{R}, \forall i \in [m]$.
- **Step 3**: The learner incurs an instantaneous loss $f_t(\theta_t, \mathcal{D}_t)$ and m fairness notions $g_i(\theta_t, \mathcal{D}_t), \forall i \in [m]$.
- **Step 4**: Advance to $t + 1$.

The goal of fairness-aware online learning [22, 28] is to (1) minimize the loss regret over the rounds, which is to compare to the cumulative loss of the best-fixed model in hindsight and (2)

ensure the total violation of fair constraints sublinearly increase in T. The loss regret is typically referred to as *static regret* since the comparator is time-invariant. To control bias and ensure group fairness across different protected sub-populations, fairness notions are considered as constraints on optimization problems.

$$
\min_{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_T \in \Theta} \quad \text{Regret}(T) = \sum_{t=1}^{T} f_t(\boldsymbol{\theta}_t, \mathcal{D}_t) - \min_{\boldsymbol{\theta} \in \Theta} \sum_{t=1}^{T} f_t(\boldsymbol{\theta}, \mathcal{D}_t)
$$

$$
\text{subject to} \quad \sum_{t=1}^{T} g_i(\boldsymbol{\theta}_t, \mathcal{D}_t) \leq O(T^\gamma), \quad \forall i \in [m], \quad \gamma \in (0,1) \tag{2}
$$

where the summation of fair constraints is defined as *long-term constraints* in [16]. The big O notation in the constraint is to bound the total violation of fairness sublinear in T. The main drawback of using the metric of static regret is that it is only meaningful for stationary environments, and low static regret cannot imply a good performance in changing environments since the time-invariant comparator in Equation (2) may behave badly [23].

4 METHODOLOGY

4.1 Settings and Problem Formulation

To address the limitation of changing environments in online learning, AR based on [10] is defined as the maximum static regret over any contiguous intervals. However, AR does not respect short intervals well. To this end, SAR [3] is proposed to improve AR, which emphasizes the dependence on lengths of intervals, and it takes the form that

$$
\text{SAR}(T, \tau) = \max_{[s, s+\tau-1] \subseteq [T]} \left(\sum_{t=s}^{s+\tau-1} f_t(\boldsymbol{\theta}_t, \mathcal{D}_t) - \min_{\boldsymbol{\theta} \in \Theta} \sum_{t=s}^{s+\tau-1} f_t(\boldsymbol{\theta}, \mathcal{D}_t) \right), \tag{3}
$$

where τ indicates the length of time interval. In SAR, the learner is competing with changing comparators, as $\boldsymbol{\theta}$ varies with s over $[s, s+\tau-1]$.

In this article, we consider the online meta-learning setting similar in [6, 25, 28], but tasks are sampled from heterogeneous distributions. Instead of static regret, we define a novel regret Fair-SAR in Equation (4). Let $\{\boldsymbol{\theta}_t\}_{t=1}^{T}$ be the sequence of model parameters generated in the *Step 1* of the learning protocol (see Section 3.3). The goal of our problem is to minimize FairSAR under the long-term fair constraints:

$$
\text{FairSAR}(T, \tau) = \max_{[s, s+\tau-1] \subseteq [T]} \left(\sum_{t=s}^{s+\tau-1} f_t\left(\mathcal{G}_t\left(\boldsymbol{\theta}_t, \mathcal{D}_t^S\right), \mathcal{D}_t^V\right) - \min_{\boldsymbol{\theta} \in \Theta} \sum_{t=s}^{s+\tau-1} f_t\left(\mathcal{G}_t\left(\boldsymbol{\theta}, \mathcal{D}_t^S\right), \mathcal{D}_t^V\right) \right)
$$

$$
\text{subject to} \quad \max_{[s, s+\tau-1] \subseteq [T]} \left(\sum_{t=s}^{s+\tau-1} g_i\left(\mathcal{G}_t\left(\boldsymbol{\theta}_t, \mathcal{D}_t^S\right), \mathcal{D}_t^V\right) \right) \leq O(T^\gamma), \quad \forall i \in [m] \tag{4}
$$

where $\gamma \in (0,1)$. $\mathcal{D}_t^S, \mathcal{D}_t^V \subset \mathcal{D}_t$ are the support and validation set. $\mathcal{G}_t(\cdot)$ is the base learner which corresponds to one or multiple gradient steps [5]. Different from traditional online learning settings, the long-term constraint violation $g(\cdot) : \mathcal{B} \times \mathbb{R}^d \to \mathbb{R}$ is satisfied. To facilitate our analysis, $\boldsymbol{\theta}_t$ is originally chosen from its domain $\Theta = \{\boldsymbol{\theta} \in \mathbb{R}^d : g_i(\boldsymbol{\theta}, \mathcal{D}_t) \leq 0, \forall i \in [m]\}$. A projection operator is hence typically applied to the updated variables to make them feasible [11, 16, 22]. To lower the computational complexity and accelerate the online processing speed, we relax the domain Θ to $\mathcal{B}$, where $\Theta \subseteq \mathcal{B} = S\mathbb{K}$ with $\mathbb{K}$ being the unit ℓ_2 ball centered at the origin, and $S = \max\{r > 0 : r = ||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||_2, \forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \Theta\}$.
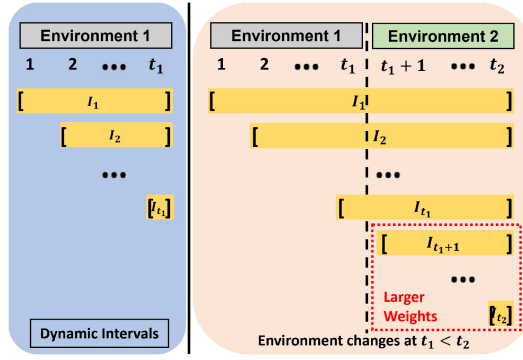
Fig. 1. An illustration of adapting to changing environments using DI. **(Left)** At time $t_1 \in [T]$, a number of intervals $\{I_1, \ldots, I_{t_1}\}$ are selected from the interval set $\mathcal{I}_{DI}$. **(Right)** At time $t_2 > t_1$, a different interval set is selected. Assume that the environment changes at t_1, to adapt to the change quickly, larger weights are given to the outputs through interval-level experts, where such outputs are not based on intervals prior to t_1.

In the protocol stated in Section 3.3, the key step (*Step 1*) is to find a good parameter θ_t at each time t. In the following subsections, we first introduce three types of intervals where each interval combines a list of tasks (Section 4.2); then, for each interval, a learning process (an expert) is proposed to output an interval-level model parameter (Section 4.3); finally, θ_t is estimated using a meta-algorithm in which it combines weighted interval-level model parameters (Section 4.4).

4.2 Intervals

In Equation (4), FairSAR evaluates the learner's performance on each time interval, and it is the maximum regret over any contiguous intervals. This subsection introduces three alternative interval sets to adapt to changing environments: **dynamic intervals** (**DI**) and two geometric covering-based intervals (AGC and DGC). Each interval in an interval set refers to a range of time indices associated with a collection of data batches, as data batches arrive one after another over time. Inspired by *learning with expert advice* [12], each interval is built upon a learning process, defined as an *expert*, and each expert updates model parameters via $\mathcal{G}$ and outputs interval-level parameters with respect to a specific interval. Details of the interval-level learning are given in Section 4.3.1.

4.2.1 Dynamic Intervals (DI). A heuristic method in designing an effective online learning algorithm for changing environments is to initiate a set of intervals $\mathcal{I}_{DI}$ dynamically, where

$$\mathcal{I}_{DI} = \bigcup_{k \in [T]} \mathcal{I}_k \quad \text{where} \quad \mathcal{I}_k = \{I_k | I_k = [k, q], \forall q \in \{k, \ldots, T\}\}. \tag{5}$$

An interval $I_k \in \mathcal{I}_{DI}$ refers to a collection of time indices $\{k, \ldots, q\}$ associated with corresponding data batches $\{\mathcal{D}_i\}_{i=k}^{q}$.

Furthermore, at each time t, we introduce a target set $C_t^{DI} \subset \mathcal{I}_{DI}$ which includes a set of intervals in $\mathcal{I}_{DI}$.

$$C_t^{DI} = \{I_i | I_i = [i, t], I_i \in \mathcal{I}_{DI}, \forall i \in [t]\}. \tag{6}$$

C_t^{DI} dynamically selects a subset of intervals from $\mathcal{I}_{DI}$. An example is illustrated in Figure 1. At time t_1, the target set $C_{t_1}^{DI}$ selects t_1 intervals where $C_{t_1}^{DI} = \{[1, t_1], [2, t_1], \ldots, [t_1, t_1]\}$. Similarly, when at time t_2 where $t_2 > t_1$, $C_{t_2}^{DI} = \{[1, t_2], [2, t_2], \ldots, [t_1, t_2], \ldots, [t_2, t_2]\}$.

Fig. 2. **(Upper)** A graphical illustration of AGC intervals (base=2) with $T = 18$. The interval set $\mathcal{I}_{AGC}$ consists of 4 subsets $\{\mathcal{I}_0, \mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3\}$ and each contains different numbers of intervals with fixed length. Intervals covered by shadow is an example of target subset C_5^{AGC} when $t = 5$. **(Lower)** An illustration of DGC intervals (base=2) with T is unknown in advance. The interval subsets $\{\mathcal{I}_0, \mathcal{I}_1, \dots\}$ increase as t increases. Similar to the setting of AGC intervals, when $t = 5$, the target set C_5^{DGC} only includes one interval I_0^5.

To adapt to changing environments, at each time, a number of experts are initiated based on intervals selected in the target set. At time t_2, each expert corresponding to an interval $I_j \in C_{t_2}^{DI}$, where $I_j = [j, t_2], \forall j \in [t_2]$, takes the parameter θ_{t_2-1} as well as its corresponding dataset $\{\mathcal{D}_i\}_{i=j}^{t_2}$ as input. Each expert independently gives an interval-level solution θ_{t_2, I_j} on I_j. A good θ_{t_2} is, therefore, achieved at time t_2 by further combining the decisions $\{\theta_{t_2, I_j}\}_{j=1}^{t_2}$ through weighted average. More details are stated in Section 4.3.

The key idea of constructing DI is that at time t_2, some of the outputs $\{\theta_{t_2, I_j}\}_{j=t_1+1}^{t_2}$ on intervals $\{I_{t_1+1}, \dots, I_{t_2}\}$ are not based on any data prior to time t_1 where $t_1 < t_2$, so that if the environment changes at t_1, those outputs may be given a larger weight by the meta-algorithm, allowing it to adapt more quickly to the change.

A main drawback with the construction of DI, however, is a factor of t increase in the time complexity. The number of intervals and learning processes increases linearly in time. To avoid this, we reduce the complexity to $O(\log t)$ by restarting algorithms on a designed set of geometric covering intervals, i.e., AGC and DGC intervals in Section 4.2.2 and 4.2.3, respectively.

*4.2.2 **Adaptive Geometric Covering (AGC) Intervals.*** Inspired by the seminal work of SAR [3], given the total number of time T, we improve DI by constructing a number of interval sets where each set $\mathcal{I}_k \subset \mathcal{I}_{AGC}$ contains various intervals with fixed lengths. We name them AGC intervals. A set of contiguous AGC intervals $\mathcal{I}_{AGC}$ are defined as

$$\mathcal{I}_{AGC} = \bigcup_{k \in [\lfloor \log_2^T \rfloor - 1] \cup \{0\}} \mathcal{I}_k \quad \text{where} \quad \forall k, \ \mathcal{I}_k = \left\{ I_k^i | I_k^i = \left[(i-1) \cdot 2^k + 1, \min\{T, i \cdot 2^k\} \right], \forall i \in \mathbb{N} \right\}.$$

(7)

An example with $T = 18$ is given in Figure 2 to illustrate the composition of AGC intervals. With selecting 2 as the log base in Equation (7), intervals are hence decomposed into $\lfloor \log_2^{18} \rfloor = 4$ subsets (i.e., $\mathcal{I}_0, \mathcal{I}_1, \mathcal{I}_2$, and $\mathcal{I}_3$) with fixed lengths of $2^0 = 1, 2^1 = 2, 2^2 = 4$, and $2^3 = 8$. Notice that the log base

Table 2. Comparison of Key Different Interval Settings

	DI	AGC	DGC
Require T in advance?	No	Yes	No
Interval lengths in $\mathcal{I}_k$	$[T-k+1]$	2^k	2^k
# total experts $\|\mathcal{U}_t\|$ at time t	t	$\lfloor \log_2^T \rfloor$	$\lfloor \log_2^t \rfloor + 1$
# active experts $\|\mathcal{A}_t\|$ at time t	t	$\leq \lfloor \log_2^T \rfloor$	$\leq \lfloor \log_2^t \rfloor + 1$
# sleeping experts $\|\mathcal{S}_t\|$ at time t	0	$< \lfloor \log_2^T \rfloor$	$< \lfloor \log_2^t \rfloor + 1$
Complexity	$\mathcal{O}(t)$	$\mathcal{O}(\log t)$	$\mathcal{O}(\log t)$

equals 2 is not required, but a larger base number leads to fewer interval subsets. AGC intervals can be considered a special case of a more general set of intervals, and they efficiently reduce the time complexity to $\mathcal{O}(\log t)$.

Similar to DI, a target set $C_t^{AGC} \subset \mathcal{I}_{AGC}$ including a set of intervals starting from t is selected from $\mathcal{I}_{AGC}$ at each time:

$$C_t^{AGC} = \{I | I \in \mathcal{I}_{AGC}, t \in I, (t-1) \notin I\}. \tag{8}$$

As shown in Figure 2, given $T = 18$ and when $t = 5$, the target set C_5^{AGC} contains three intervals, $[5, 5]$, $[5, 6]$ and $[5, 8]$, where each initiates at $t = 5$ with interval lengths 1, 2, and 4, respectively.

4.2.3 ***Dynamic Geometric Covering (DGC)*** *Intervals.* Although the setting of AGC intervals efficiently reduces the complexity, one limitation is that the total number of times T needs to be known and fixed in advance. However, this assumption does not always hold. This leads to the number of interval sets (i.e., $\lfloor \log_2^T \rfloor$) being unchanged in AGC, as $k \in \{0, \dots, \lfloor \log_2^T \rfloor - 1\}$.

To tackle this limitation, we alternatively propose another type of interval set, namely, DGC intervals, $\mathcal{I}_{DGC}$.

$$\mathcal{I}_{DGC} = \bigcup_{k \in \{0,1,2,\dots\}} \mathcal{I}_k \quad \text{where} \quad \forall k, \ \mathcal{I}_k = \left\{ I_k^i | I_k^i = \left[i \cdot 2^k, (i+1) \cdot 2^k - 1 \right], \forall i \in \mathbb{N} \right\}. \tag{9}$$

Figure 2 illustrates the difference between the settings of DGC and AGC intervals. Since the total number of times is unknown in advance, the number of interval sets, $\lfloor \log_2 t \rfloor + 1$, in DGC increases as t becomes larger. For each interval set $\mathcal{I}_k$, its first interval $I_k^1 \in \mathcal{I}_k$ initializes at the 2^k-th time, and each interval $I_k^i \in \mathcal{I}_k$ holds the same length of 2^k. Furthermore, the setting of the target set $C_t^{DGC} \subset \mathcal{I}_{DGC}$ at time t is the same as the one in AGC, referring to Equation (8). As indicated in Figure 2, in contrast to AGC, in DGC, an additional interval set $\mathcal{I}_4$ is initialized at time $t = 16$ with an interval length of $2^4 = 16$. Similarly, when $t = 5$, the target set C_5^{DGC} includes the interval $[5, 5]$ only, as there is one interval in $\mathcal{I}_{DGC}$ that starts at time 5.

A brief comparison between different interval settings introduced in this section is listed in Table 2.

4.3 Learning Experts

4.3.1 The Interval-Level Learning within an Expert. As we mentioned at the beginning of Section 4.2, each interval is built upon a learning process, defined as an expert. At time t, an expert E is a learning algorithm $\mathcal{G} : \Theta \times \mathbb{R}^d \rightarrow \Theta$ (a.k.a., a base learner, such as one or multiple gradient steps [5]) within an interval that inputs parameters θ_{t-1} and outputs interval-level parameters $\theta_{t,I}$ specific to the interval I. The *interval-level* parameter update for an expert E on interval I at time t is defined

$$\theta_{t,I} := \mathcal{G}_t \left(\theta, \mathcal{D}_{t,I}^S \right) = \arg\min_{\theta} \ f_t \left(\theta, \mathcal{D}_{t,I}^S \right) \quad \text{subject to} \quad g_i \left(\theta, \mathcal{D}_{t,I}^S \right) \leq 0, \forall i \in [m], \tag{10}$$

where the loss function $f_t(\cdot)$ and the fairness function $g_i(\cdot)$ are defined based on the support set $\mathcal{D}_{t,I}^S \subset \mathcal{D}_{t,I}$ associated with E.

4.3.2 Active and Sleeping Experts. Inspired by *learning with expert advice* problems [12], we dynamically construct a set of experts $\mathcal{U}_t = \{E_k\}_{k=0}^{|\mathcal{U}_t|}$ at each time t. Recall that we introduce three types of interval sets in Section 4.2. The number of total experts for each setting is various, where at time t

  — for DI, $|\mathcal{U}_t^{DI}| = t$. The number of experts increases by 1 at each time.
  — for AGC intervals, $|\mathcal{U}_t^{AGC}| = \lfloor \log_2 T \rfloor$. The number of experts is unchanged at different times, resulting from T being known in advance and fixed.
  — for DGC intervals, $|\mathcal{U}_t^{DGC}| = \lfloor \log_2 t \rfloor + 1$. The number of experts slowly increases as needed, without known T in advance.

Furthermore, to adapt to changing environments efficiently, all experts are dynamically partitioned into active and sleeping (or inactive) experts at each time t, denoted $\mathcal{A}_t \subseteq \mathcal{U}_t$ and $\mathcal{S}_t = \mathcal{U}_t \backslash \mathcal{A}_t$, respectively. As indicated in Section 4.2, a target set C_t, for all types of intervals $\mathcal{I}_{DI}$, $\mathcal{I}_{AGC}$, and $\mathcal{I}_{DGC}$, is subsetted from the interval set. Active experts are experts corresponding to intervals in the target sets, wherein active experts update model parameters at interval-level using Equation (10). For sleeping experts, as no corresponding intervals are selected in the target set at time t, their interval-level model parameters are not updated and remain at the last update. Similarly, the number of active/sleeping experts varies by applying different types of interval sets, where at time t

  — for DI, all experts are active experts $\mathcal{A}_t^{DI} = \{E_k\}_{k=1}^t$ and the number of sleeping experts is zero, $\mathcal{S}_t^{DI} = \emptyset$.
  — for AGC and DGC intervals, the number of active experts is the cardinality of the selected target set. As the example shown in Figure 2, when $t = 5$, active experts are $\mathcal{A}_5^{AGC} = \{E_0, E_1, E_2\}$ and $\mathcal{A}_5^{DGC} = \{E_0\}$, and sleeping experts are $\mathcal{S}_5^{AGC} = \{E_3\}$ and $\mathcal{S}_5^{DGC} = \{E_1, E_2\}$ (experts E_3 and E_4 are not initialized until $t = 8$ and $t = 16$, respectively).

4.4 Learning Dynamically for Bi-Level Adaptation

Recall that in the protocol of fairness-aware online learning (Section 3.3), the main goal for the learner is to sequentially decide on the model parameter θ_t that performs well on the loss sequence and the long-term fair constraints. Crucially, inspired by [5], we consider a setting where at each round t the learner can perform a number of expert-specific updates at an interval level in the active set $\mathcal{A}_t$.

As specified in Equation (4), model parameters at each round t are determined by formulating problems with a nested bi-level adaptation process: interval-level and meta-level. Each level corresponds to a sub-learner, i.e., base and meta learner, respectively, described in Figure 3. The problem of learning a meta-level parameter θ_t is embedded with the optimization problem of finding interval-level parameters $\theta_{t,I}$ in Equation (10). For experts in the sleeping set $\mathcal{S}_t$, the base learner is not applied. The *meta-level* problem takes the form in Equation (11).

$$
\begin{aligned}
\min_{\theta \in \mathcal{B}} \quad & \sum_{E_k \in \mathcal{A}_t} p_{t,I_k} \cdot f_t\left(\mathcal{G}_t\left(\theta, \mathcal{D}_{t,I_k}^S\right), \mathcal{D}_{t,I_k}^Q\right) + \sum_{E_k \in \mathcal{S}_t} p_{t,I_k} \cdot f_t\left(\theta_{t',I_k}, \mathcal{D}_{t,I_k}^Q\right) \\
\text{subject to} \quad & \sum_{E_k \in \mathcal{A}_t} p_{t,I_k} \cdot g_i\left(\mathcal{G}_t\left(\theta, \mathcal{D}_{t,I_k}^S\right), \mathcal{D}_{t,I_k}^Q\right) + \sum_{E_k \in \mathcal{S}_t} p_{t,I_k} \cdot g_i\left(\theta_{t',I_k}, \mathcal{D}_{t,I_k}^Q\right) \le 0
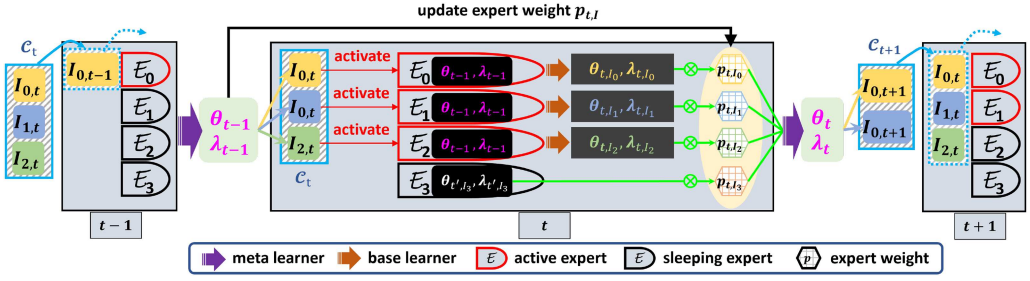\end{aligned}
\tag{11}
$$

Fig. 3. An overview of FairSAOML with AGC or DGC intervals to determine model parameter pair at each round. A target set (shadowed) of intervals is initially selected and is later used to activate corresponding experts. Each active expert runs through a base learner for the interval-level parameter-pair adaption, and its weight is updated. The meta-level parameter pair is finally attained through the meta-learner by combining the weighted actions of all experts.

where $p_{t,I_k} \geq 0$ is the expert weight of E_k at t. $\mathcal{D}^Q_{t,I_k} \subset \mathcal{D}_{t,I_k}$ is the query set where $\mathcal{D}^Q_{t,I_k} \cap \mathcal{D}^S_{t,I_k} = \emptyset$. θ_{t',I_k} is the interval-level model parameter for a sleeping expert $E_k \in \mathcal{S}_t$ where the round index $t' < t$ represents the last time this expert was activated.

In the following section, we introduce our proposed algorithm FairSAOML. Instead of optimizing primal parameters only, it efficiently deals with the bi-level optimization problem of Equations (10) and (11) by approximating a sequence of pairs of primal-dual meta parameters $\{(\theta_t, \lambda_t)\}_{t=1}^T$ where the pair respectively responds for adjusting accuracy and fairness level.

4.5 An Efficient Algorithm: FairSAOML

To find a good model parameter pair (θ_t, λ_t) at each time, an efficient working flow is proposed in Algorithm 1. Inspired by dynamic programming and expert-tracking [15] techniques, experts at each time are recursively divided into active and sleeping ones. Model parameters in active experts are locally updated, but those in sleeping experts are directly inherited from the previous time. Specifically, at the beginning of t, a target set C_t containing intervals is used to activate a subset of experts in $\mathcal{U}_t$. For each active expert E_k in $\mathcal{A}_t$, an interval-level algorithm takes the meta-level solution $(\theta_{t-1}, \lambda_{t-1})$ and outputs an expert-specific solution pair $(\theta_{t,I_k}, \lambda_{t,I_k})$. Finally, through the meta-learner, we combine the weighted solutions of all experts and move to the next time.

We explain the main steps in Algorithm 1 below. In Step 4, when a new task arrives at time t, a batch of data $\mathcal{D}^V_t$ is randomly sampled from $\mathcal{D}_t$ for validation purposes, and the performance on θ_{t-1} achieved is recorded. A target set of intervals C_t is selected from $\mathcal{I}$ in Step 5. For each interval $I_k \in C_t$ (Step 6–8), the corresponding expert E_{t,I_k} is activated, according to a specific ACTIVATE-EXPERTS procedure on the choice of interval sets indicated in the subroutines of FairSAOML in Algorithm 2.

We present three distinct expert activation procedures in Algorithm 2. For each active expert, we set adaptive stepsizes $\eta_{t,I} = S/(G\sqrt{|I_k|})$, where S is the radius of the Euclidean ball $\mathcal{B}$, and there exists a constant $G > 0$ that bounds the (sub)gradients of f_t and g_i. Following the setting used in [11], empirically we set $S = \sqrt{1 + 2\epsilon} - 1$ and $G = \max\{\sqrt{d} + S, \max_t\{||\boldsymbol{e}_{I_k}||_2, \boldsymbol{e}_{I_k} \in \mathcal{P}_t\}\}$, where $\boldsymbol{e}_{I_k}$ is the non-protected features lied in the interval I_k and d is its feature dimension. $\mathcal{P}_t$ is a set which includes all past intervals until time t. Specifically in DI and DGC, at some time t, new experts are initiated. We set the constants R_{t,I_k} and C_{t,I_k} to zeros that are further used to change the corresponding expert weight to adapt to changing environments.

ALGORITHM 1: FairSAOML

  1: Initialize meta-parameters pair $(\boldsymbol{\theta}_0, \boldsymbol{\lambda}_0)$, where $\boldsymbol{\theta}_0$ is the center of $\mathcal{B}$ and $\boldsymbol{\lambda}_0 \in \mathbb{R}_+^m$ is randomly chosen
  2: Create an object-oriented expert E containing interval level parameter pair $(\boldsymbol{\theta}_I, \boldsymbol{\lambda}_I)$, learning rate η_I, constants R_I, C_I
  3: **for** each $t \in [T]$ **do**
  4:     Sample $\mathcal{D}_t^V \subset \mathcal{D}_t$ and record the performance of $\boldsymbol{\theta}_{t-1}$
  5:     Subset C_t from $\mathcal{I}$ using either Equation (6) or Equation (8).
  6:     **for** each interval $I_k \in C_t$ **do**
  7:         ACTIVATEEXPERTS(DI or AGC or DGC)
  8:     **end for**
  9:     **for** each expert $E_k \in \mathcal{U}_t$ **do**
 10:         Update p_{t,I_k} using R_{t,I_k}, C_{t,I_k} in Equation (12)
 11:     **end for**
 12:     **for** $n = 1, \dots, N_{meta}$ steps **do**
 13:         **for** each active expert $E_k \in \mathcal{A}_t$ **do**
 14:             Sample support set $\mathcal{D}_{t,I_k}^S \subset \mathcal{D}_{t,I_k}$
 15:             Adapt interval-level primal and dual variables with $\mathcal{D}_{t,I_k}^S$ using Equation (14)
 16:         **end for**
 17:         **for** each expert $E_k \in \mathcal{U}_t$ **do**
 18:             Sample query set $\mathcal{D}_{t,I_k}^Q \subset \mathcal{D}_{t,I_k}$
 19:         **end for**
 20:         Update meta-level primal and dual variables with D_{t,I_k}^Q using Equation (16)
 21:     **end for**
 22:     **for** each expert $E_k \in \mathcal{U}_t$ **do**
 23:         $R_{t+1,I_k} = R_{t,I_k} + \mathcal{F}_{t,I_k}(\boldsymbol{\theta}_t, \boldsymbol{\lambda}_t) - \mathcal{F}_{t,I_k}(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k})$
 24:         $C_{t+1,I_k} = C_{t,I_k} + |\mathcal{F}_{t,I_k}(\boldsymbol{\theta}_t, \boldsymbol{\lambda}_t) - \mathcal{F}_{t,I_k}(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k})|$
 25:     **end for**
 26: **end for**

In Steps 9–11 of Algorithm 1, for all experts in $\mathcal{U}_t$, a following weight p_{t,I_k} is estimated:

$$p_{t,I_k} = \frac{w(R_{t,I_k}, C_{t,I_k})}{\sum_{E_k \in \mathcal{U}_t} w(R_{t,I_k}, C_{t,I_k})}. \tag{12}$$

Here, a weight function [15] is defined as $w(R, C) = \frac{1}{2}\big(\Phi(R + 1, C + 1) - \Phi(R - 1, C - 1)\big)$, where $\Phi(R, C) = \exp([R]_+^2/3C)$ and $[r]_+ = \max(0, r)$ and $\Phi(0, 0) = 1$. In Steps 12–21, our FairSAOML responds to the bi-level adaptation stated in Equations (10) and (11). Specifically, to solve the interval-level problem in Equation (10), for each active expert E_k in $\mathcal{A}_t$, we consider following Lagrangian function

$$\mathcal{F}_{t,I_k}(\boldsymbol{\theta}_{t-1}, \boldsymbol{\lambda}_{t-1}) = f_t\left(\boldsymbol{\theta}_{t-1}, \mathcal{D}_{t,I_k}^S\right) + \sum_{i=1}^m \lambda_{t-1,i} \cdot g_i\left(\boldsymbol{\theta}_{t-1}, \mathcal{D}_{t,I_k}^S\right), \tag{13}$$

where the interval-level parameter pair for an active expert E_k are initialized with the meta-level parameter $(\boldsymbol{\theta}_{t-1}, \boldsymbol{\lambda}_{t-1})$. For optimization with simplicity, cumulative constraints in Equation (10) are approximated with the summarized regularization. Interval-level parameters are updated through a base learner $\mathcal{G}_t(\cdot)$. One example for the learner is updating with one gradient step [5] using the pre-determined adaptive stepsize η_{t,I_k}. Notice that for multiple gradient steps, $\boldsymbol{\theta}_{t,I_k}$ and

---

ALGORITHM 2: FairSAOML Subroutines

1: **Require**: Choose one of the three settings of intervals.
2: **procedure** ACTIVATEEXPERTS(DI)
3:     Activate expert E_k by letting $\eta_{t,I_k} = S/(G\sqrt{|I_k|})$, $(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k}) \leftarrow (\boldsymbol{\theta}_{t-1}, \boldsymbol{\lambda}_{t-1})$
4:     **if** Identify an $E_j \in \mathcal{U}_t$ where $|I_j| = |I_k| - 1$ **then**
5:         Update $R_{t,I_k} \leftarrow R_{t-1,I_j}$ and $C_{t,I_k} \leftarrow C_{t-1,I_j}$
6:         Replace E_j with E_k in $\mathcal{U}_t$
7:     **else**
8:         Set $R_{t,I_k} = 0$ and $C_{t,I_k} = 0$.
9:         Add the E_k to $\mathcal{U}_t$
10:     **end if**
11: **end procedure**
12: **procedure** ACTIVATEEXPERTS(AGC)
13:     Activate expert E_k by letting $\eta_{t,I_k} = S/(G\sqrt{|I_k|})$, $(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k}) \leftarrow (\boldsymbol{\theta}_{t-1}, \boldsymbol{\lambda}_{t-1})$
14:     Identify an $E_j \in \mathcal{U}_t$ where $|I_j| = |I_k|$
15:     Update $R_{t,I_k} \leftarrow R_{t-1,I_j}$ and $C_{t,I_k} \leftarrow C_{t-1,I_j}$
16:     Replace E_j with E_k in $\mathcal{U}_t$
17: **end procedure**
18: **procedure** ACTIVATEEXPERTS(DGC)
19:     Activate expert E_k by letting $\eta_{t,I_k} = S/(G\sqrt{|I_k|})$, $(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k}) \leftarrow (\boldsymbol{\theta}_{t-1}, \boldsymbol{\lambda}_{t-1})$
20:     **if** Identify an $E_j \in \mathcal{U}_t$ where $|I_j| = |I_k|$ **then**
21:         Update $R_{t,I_k} \leftarrow R_{t-1,I_j}$ and $C_{t,I_k} \leftarrow C_{t-1,I_j}$
22:         Replace E_j with E_k in $\mathcal{U}_t$
23:     **else**
24:         Set $R_{t,I_k} = 0$ and $C_{t,I_k} = 0$.
25:         Add the E_k to $\mathcal{U}_t$
26:     **end if**
27: **end procedure**

$\boldsymbol{\lambda}_{t,I_k}$ interplay each other for updating.

$$\boldsymbol{\theta}_{t,I_k} = \boldsymbol{\theta}_{t-1} - \eta_{t,I_k}\nabla_{\boldsymbol{\theta}}\mathcal{F}_{t,I_k}(\boldsymbol{\theta}_{t-1}, \boldsymbol{\lambda}_{t-1}); \quad \boldsymbol{\lambda}_{t,I_k} = \boldsymbol{\lambda}_{t-1} + \eta_{t,I_k}\nabla_{\boldsymbol{\lambda}}\mathcal{F}_{t,I_k}(\boldsymbol{\theta}_{t,I}, \boldsymbol{\lambda}_{t-1}). \quad (14)$$

Next, to solve the meta-level problem in Equation (11), we combine the actions of active experts together with sleeping experts. We consider the following augmented Lagrangian function and abuse the symbol t' with t in Equation (11):

$$\mathcal{L}_t(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k}) = \sum_{E_k \in \mathcal{U}_t} p_{t,I_k} \left(f_t\left(\boldsymbol{\theta}_{t,I_k}, \mathcal{D}^Q_{t,I_k}\right) + \sum_{i=1}^{m}\left(\lambda_{i,t,I_k} \cdot g_i\left(\boldsymbol{\theta}_{t,I_k}, \mathcal{D}^Q_{t,I_k}\right) - \frac{\delta(\eta_1 + \eta_2)}{2}\lambda^2_{i,t,I_k}\right)\right),$$
$$(15)$$

where $\delta > 0$ is a constant determined by analysis. Note that the last augmented term on the dual variable is devised to prevent $\boldsymbol{\lambda}$ from being too large. The update rule for meta-level parameters follows:

$$\boldsymbol{\theta}_t = \prod_{\mathcal{B}}\left(\boldsymbol{\theta}_{t-1} - \eta_1\nabla_{\boldsymbol{\theta}}\mathcal{L}_t(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k})\right); \quad \boldsymbol{\lambda}_t = \left[\boldsymbol{\lambda}_{t-1} + \eta_2\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\boldsymbol{\theta}_{t,I_k}, \boldsymbol{\lambda}_{t,I_k})\right]_+, \quad (16)$$

where $\prod_{\mathcal{B}}$ is the projection operation to the relaxed domain $\mathcal{B}$ that is introduced in Section 4.1. This approximates the true desired projection with a simpler closed form. Finally, in Steps 22–25, we update each expert's R and C values, determining the expert weight for the next time. The

intuition of weight update is to re-adjust the difference between the meta-solution and the interval-level solution given by the expert.

5 ANALYSIS

To analyze, we first make the following assumptions as in [16, 23]. Examples where these assumptions hold include logistic regression and L_2 regression over a bounded domain. As for constraints, a family of fairness notions, such as DDP stated in Definition 1, are applicable as discussed in [14]. For simplicity, in this section we omit $\mathcal{D}$ used in $f_t(\cdot), \forall t$ and $g_i(\cdot), \forall i$.

ASSUMPTION 1 (CONVEX DOMAIN). *The convex set Θ is non-empty, closed, bounded, and it is described by m convex functions as $\Theta = \{\boldsymbol{\theta} : g_i(\boldsymbol{\theta}) \leq 0, \forall i \in [m]\}$. The relaxed domain $\mathcal{B}$ (where $\Theta \subseteq \mathcal{B}$) contains the origin $\mathbf{0}$ and its diameter is bounded by S.*

ASSUMPTION 2. *Both the loss functions $f_t(\cdot), \forall t$ and constraint functions $g_i(\cdot), \forall i \in [m]$ satisfy the following assumptions*

(1) *(Lipschitz Continuous) $\forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \mathcal{B}, ||f_t(\boldsymbol{\theta}_1) - f_t(\boldsymbol{\theta}_2)|| \leq L_f||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||, ||g_i(\boldsymbol{\theta}_1) - g_i(\boldsymbol{\theta}_2)|| \leq L_g||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||$. Let $G = \max\{L_f, L_g\}$, $F = \max_{t \in [T]} \max_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \mathcal{B}} f_t(\boldsymbol{\theta}_1) - f_t(\boldsymbol{\theta}_2) \leq 2L_f S$, and $D = \max_{i \in [m]} \max_{\boldsymbol{\theta} \in \mathcal{B}} g_i(\boldsymbol{\theta}) \leq L_g S$.*
(2) *(Lipschitz Gradient) $f_t(\boldsymbol{\theta}), \forall t$ are β_f-smooth and $g_i(\boldsymbol{\theta}), \forall i$ are β_g-smooth, that is, $\forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \mathcal{B}$, $||\nabla f_t(\boldsymbol{\theta}_1) - \nabla f_t(\boldsymbol{\theta}_2)|| \leq \beta_f||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||, ||\nabla g_i(\boldsymbol{\theta}_1) - \nabla g_i(\boldsymbol{\theta}_2)|| \leq \beta_g||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||$.*
(3) *(Lipschitz Hessian) Twice-differentiable functions $f_t(\boldsymbol{\theta}), \forall t$ and $g_i(\boldsymbol{\theta}), \forall i$ have ρ_f and ρ_g- Lipschitz Hessian, respectively. That is, $\forall \boldsymbol{\theta}_1 - \boldsymbol{\theta}_2 \in \mathcal{B}, ||\nabla^2 f_t(\boldsymbol{\theta}_1) - \nabla^2 f_t(\boldsymbol{\theta}_2)|| \leq \rho_f||\boldsymbol{\theta} - \boldsymbol{\phi}||, ||\nabla^2 g_i(\boldsymbol{\theta}_1) - \nabla^2 g_i(\boldsymbol{\theta}_2)|| \leq \rho_g||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||$.*

ASSUMPTION 3 (STRONGLY CONVEXITY). *Suppose $f_t(\boldsymbol{\theta}), \forall t$ and $g_i(\boldsymbol{\theta}), \forall i$ have strong convexity, that is, $\forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \mathcal{B}, ||\nabla f_t(\boldsymbol{\theta}_1) - \nabla f_t(\boldsymbol{\theta}_2)|| \geq \mu_f||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||, ||\nabla g_i(\boldsymbol{\theta}_1) - \nabla g_i(\boldsymbol{\theta}_2)|| \geq \mu_g||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||$.*

Under the above assumptions, we first state the key Theorem 1 that the proposed FairSAOML enjoys a sub-linear guarantee for both regret and long-term fairness constraints in the long run for Algorithm 1.

THEOREM 1. *Set $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta} \in \Theta} \sum_{t=s}^{s+\tau-1} f_t(\mathcal{G}_t(\boldsymbol{\theta}))$ where $[s, s + \tau - 1] \subseteq [T]$. Under Assumptions 1, 2, and 3, the regret FairSAR proposed in Equation (4) of FairSAOML in Algorithm 1 satisfies the bounds in Equation (17) for all three interval settings that stated in Section 4.2.*

$$
\max_{[s, s+\tau-1] \subseteq [T]} \left(\sum_{t=s}^{s+\tau-1} f_t\Big(\mathcal{G}_t(\boldsymbol{\theta}_t)\Big) - f_t\Big(\mathcal{G}_t(\boldsymbol{\theta}^*)\Big) \right) \leq O\Big((\tau \log T)^{1/2}\Big)
$$
$$
\max_{[s, s+\tau-1] \subseteq [T]} \left(\sum_{t=s}^{s+\tau-1} g_i\Big(\mathcal{G}_t(\boldsymbol{\theta}_t)\Big) \right) \leq O\Big((\tau T \log T)^{1/4}\Big), \quad \forall i \in [m]
$$

(17)

Under Assumptions 1, 2, and 3, we target Equation (15) and have

LEMMA 1 (THEOREM 1 IN [28]). *Suppose f and $g : \Theta \times \mathbb{R}_+^m \rightarrow \mathbb{R}$ satisfy Assumptions 1, 2, and 3. The interval-level update and the augmented Lagrangian function $\mathcal{L}_t(\boldsymbol{\theta}, \boldsymbol{\lambda})$ are defined in Equations (14), (12) and Equation (15). Then, the function $\mathcal{L}_t(\boldsymbol{\theta}, \boldsymbol{\lambda})$ is convex-concave with respect to the arguments $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$, respectively. Furthermore, as for $\mathcal{L}_t(\cdot, \boldsymbol{\lambda})$, if stepsize $\eta_{t,I}$ for each active expert E_I is selected as $\eta_{t,I} \leq \min\{\frac{\mu_f + \bar{\lambda} m \mu_g}{8(L_f + \bar{\lambda} m L_g)(\rho_f + \bar{\lambda} m \rho_g)}, \frac{1}{2(\beta_f + \bar{\lambda} m \beta_g)}\}$, then $\mathcal{L}_t(\cdot, \boldsymbol{\lambda})$ enjoys $\frac{9}{8}(\beta_f + \bar{\lambda} m \beta_g)$-smooth and $\frac{1}{8}(\mu_f + \bar{\lambda} m \mu_g)$-strongly convex, where $\bar{\lambda} \geq 0$ is the mean value of $\boldsymbol{\lambda}$.*

According to Theorems 1 and 3 in [15] and the Lemma 1 in [23], we have the following lemma with respect to Equation (15) that

LEMMA 2. *Under Assumption 2, for any interval $I = [i, j] \in \mathcal{I}$, FairSAOML satisfies*

$$\sum_{u=i}^{t} \mathcal{L}_u(\theta_u, \lambda_u) - \sum_{u=i}^{t} \mathcal{L}_u(\theta_{u,I}, \lambda_{u,I}) \leq S\sqrt{6L_f L_g(t - i - 1)c(t)},$$

where $c(t) \leq 1 + \ln t + \ln(1 + \log_2^T) + \ln \frac{5+3\ln(1+t)}{2}$.

By applying Lemma 2 with the Theorem 2 in [23], we have

LEMMA 3. *Under Assumptions 1 and 2, for any interval $I = [i, j] \in \mathcal{I}$, for any $(\theta, \lambda) \in \Theta \times \mathbb{R}_+^m$ FairSAOML satisfies*

$$\sum_{t \in I} \mathcal{L}_t(\mathcal{G}_t(\theta_t), \lambda) - \sum_{t \in I} \mathcal{L}_t(\theta, \lambda_{t,I}) \leq S\sqrt{|I|}(\sqrt{6L_f L_g c(t)} + G).$$

To extend our Lemma 3 to any interval $I = [r, s] \subseteq [T]$, we refer the following lemma

LEMMA 4 (LEMMA 3 IN [23]). *For any interval $[r, s] \subseteq [T]$, it can be partitioned into two sequences of disjoint and consecutive intervals, denoted by $I_{-p}, \ldots, I_0 \in \mathcal{I}$ and $I_1, \ldots, I_q \in \mathcal{I}$, such that*

$$|I_{-i}|/|I_{-i+1}| \leq 1/2, \forall i \geq 1 \quad and \quad |I_i|/|I_{i-1}| \leq 1/2, \forall i \geq 2.$$

Finally, we prove the proposed Theorem 1.

PROOF. By applying Lemma 3 onto Lemma 4 and set θ^* being the optimal solution for $\min_{\theta \in \Theta} \sum_{t=r}^{s} f_t(\mathcal{G}_t(\theta))$ where $[r, s] \subseteq [T]$, we have

$$\sum_{t=r}^{s} \mathcal{L}_t(\mathcal{G}_t(\theta_t), \lambda) - \sum_{t=r}^{s} \mathcal{L}_t(\mathcal{G}_t(\theta^*), \lambda_{t,I}) = \sum_{i=-p}^{q} \left(\sum_{t \in I_i} \mathcal{L}_t(\mathcal{G}_t(\theta_t), \lambda) - \sum_{t \in I_i} \mathcal{L}_t(\mathcal{G}_t(\theta^*), \lambda_{t,I}) \right)$$

$$\leq \sum_{i=-p}^{q} S\sqrt{|I_i|}(\sqrt{6L_f L_g c(s)} + G) \qquad . \quad (18)$$

$$\leq 2S \left(\sqrt{6L_f L_g c(s)} + G \right) \sum_{i=0}^{\infty} \sqrt{2^{-i}|I|}$$

$$\leq 8S \left(\sqrt{6L_f L_g c(s)} + G \right) \sqrt{|I|}$$

By expanding Equation (18) using Equation (15) and following the Theorem 3.1 in [2], we have

$$\sum_{t=r}^{s} \left\{ f_t(\mathcal{G}_t(\theta_t)) - f_t(\mathcal{G}_t(\theta^*)) \right\} + \sum_{i=1}^{m} \left\{ \lambda_i \sum_{t=r}^{s} g_i(\mathcal{G}_t(\theta_t)) - \sum_{t=r}^{s} \lambda_{t,i} g_i(\mathcal{G}_t(\theta^*)) \right\}$$

$$- \frac{\delta(\eta_1 + \eta_2)(s - r + 1)}{2} ||\lambda||^2 + \frac{\delta(\eta_1 + \eta_2)}{2} \sum_{t=r}^{s} ||\lambda||^2 \leq 8S \left(\sqrt{6L_f L_g c(s)} + G \right) \sqrt{|I|}$$

Here, we approximately average p_{t,I_k} for all experts $E_k \in \mathcal{U}_t$ at time t, and hence the subscription k is omitted. Inspired by the proof of Theorem 4 in [16], we take maximization for λ over $(0, +\infty)$ and get

$$\sum_{t=r}^{s} \left\{ f_t(\mathcal{G}_t(\theta_t)) - f_t(\mathcal{G}_t(\theta^*)) \right\} + \sum_{i=1}^{m} \left\{ \frac{\left[\sum_{t=r}^{s} g_i(\mathcal{G}_t(\theta_t)) \right]_+^2}{2(\delta(\eta_1 + \eta_2)(s - r + 1) + \frac{m}{\eta_1 + \eta_2})} - \sum_{t=r}^{s} \lambda_{t,i} g_i(\mathcal{G}_t(\theta^*)) \right\}.$$

$$\leq 8S \left(\sqrt{6L_f L_g c(s)} + G \right) \sqrt{|I|}$$

Table 3. Comparison of Upper Bounds in Loss Regret and Constraint Violations across Various Methods

	Static Environment				Changing Environment		
Algorithms	FTML[6]	FairFML[28]	FairAOGD[11]	FairGLC[22]	AOD[23]	CBCE[12]	FairSAOML(Ours)
Loss Regret	$O(\log T)$	$O(\log T)$	$O(T^{2/3})$	$O(\log T)$	$O((\tau \log T)^{1/2})$	$O((\tau \log T)^{1/2})$	$O((\tau \log T)^{1/2})$
Constraint Violations	-	$O((T \log T)^{1/2})$	$O(T^{2/3})$	$O((T \log T)^{1/2})$	-	-	$O((\tau T \log T)^{1/4})$

Since $g_i(\mathcal{G}_t(\boldsymbol{\theta}^*)) \leq 0$ and $\lambda_{t,i} \geq 0, \forall i \in [m]$, the resulting inequality becomes

$$\sum_{t=r}^{s} \{f_t(\mathcal{G}_t(\boldsymbol{\theta}_t)) - f_t(\mathcal{G}_t(\boldsymbol{\theta}^*))\} + \sum_{i=1}^{m} \frac{\left[\sum_{t=r}^{s} g_i(\mathcal{G}_t(\boldsymbol{\theta}_t))\right]_+^2}{2(\delta(\eta_1 + \eta_2)(s - r + 1) + \frac{m}{\eta_1+\eta_2})} \leq 8S \left(\sqrt{6L_f L_g c(s)} + G\right) \sqrt{|I|}.$$

Due to non-negative of $\frac{\left[\sum_{t=r}^{s} g_i(\mathcal{G}_t(\boldsymbol{\theta}_t))\right]_+^2}{2(\delta(\eta_1+\eta_2)(s-r+1)+\frac{m}{\eta_1+\eta_2})}$, we have

$$\sum_{t=r}^{s} \{f_t(\mathcal{G}_t(\boldsymbol{\theta}_t)) - f_t(\mathcal{G}_t(\boldsymbol{\theta}^*))\} \leq 8S \left(\sqrt{6L_f L_g c(s)} + G\right) \sqrt{|I|} = O\left((|I| \log s)^{1/2}\right).$$

Furthermore, we have $\sum_{t=r}^{s} \{f_t(\mathcal{G}_t(\boldsymbol{\theta}_t)) - f_t(\mathcal{G}_t(\boldsymbol{\theta}^*))\} \geq -F(s - r + 1)$ according to the assumption and set $\eta_1 = \eta_2 = O(1/\sqrt{s})$. We have

$$\sum_{t=r}^{s} g_i(\mathcal{G}_t(\boldsymbol{\theta}_t)) \leq O\left((|I|s \log s)^{1/4}\right).$$

Therefore, as for FairSAR proposed in Equation (4), we complete the proof.   □

Discussion for Upper Bounds. Under aforementioned assumptions and provable convexity of Equation (18) in $\boldsymbol{\theta}$ (see Lemma 1), the proposed FairSAOML in Algorithm 1 achieves sub-linear bounds in FairSAR for both loss regret and violation of fairness constraints. Although such bounds are comparable with the strongly adapted loss regret in [12, 23] (see Table 3) in terms of online learning in changing environment paradigms, we bound loss regret and cumulative fairness constraints simultaneously. On the other hand, in terms of fairness-aware online learning, our proposed method outperforms [11, 22, 28] by giving a tighter bound of fair constraint violations.

Complexity. The computational complexity of FairSAOML in Algorithm 1 at each time $t \in [T]$ is $O(N_{meta} \cdot |\mathcal{U}_t|)$ where N_{meta} is the number of meta-level iterations and $|\mathcal{U}_t|$ is the total number of experts that need to be maintained at t, and the complexity of each expert is $O(1)$.

6 EXPERIMENTAL SETTINGS

6.1 Datasets

We use the following publicly available datasets. (1) **New York Stop-and-Frisk (NYSF)** [13] is a prominent dataset of a real-world application on policing in New York City from 2009 to 2010. It documents whether a pedestrian who was stopped on suspicion of weapon possession would in fact possess a weapon. As this data had a pronounced racial bias on African Americans, for each frisked record, we consider race as the binary protected attribute, that is black and non-black. Besides, this dataset consists of records collected in five different sub-districts, Manhattan (M), Brooklyn (B), Queens (Q), Bronx (R), and Staten (S). Since there are large performance disparities across districts and race groups, each district is viewed as an independent domain. To adapt the online learning setting, data in each domain is further split into 32 tasks and each task corresponds to ten days of a month with 111 non-protected features. According to *DDP* values in Definition 1, the fairness levels from low to high are Bronx (0.74), Queens (0.68), Staten (0.65),

Manhattan (0.53), and Brooklyn (0.44). The larger *DDP* values indicate a lower fairness level. We hence consider two settings for domain adaptation where each setting contains 96 tasks in total: (i) fairness level from high to low: Brooklyn to Manhattan to Staten (B→M→S); and (ii) fairness level from low to high: Bronx to Queens then Staten (R→Q→S). (2) *MovieLens*[1] contains 100 k ratings by 943 users on 1,682 movies, and each rating is given a binary label ("recommending" if the rating is greater than 3, "not recommending" otherwise). We consider gender as the protected attribute. To generate dynamic environments, following [18], we construct a larger dataset by combining three copies of the original data and flipping the original values of non-protected attributes by multiplying -1 for the middle copy. Therefore, each copy is considered as a data domain. Furthermore, each data copy is split into 30 tasks by timestamps, and there are 90 tasks in total.

6.2 Evaluation Metrics

Two popular evaluation metrics are introduced that each allows quantifying the extent of bias taking into account the protected attribute. DP [4] and **Equalized Odds** (**EO**) [8] can be formalized as

$$\text{DP} = k, \text{ if DP} \leq 1; \text{DP} = 1/k, \text{ otherwise,} \quad \text{where } k = \frac{\mathbb{P}(\hat{Y} = 1|S = -1)}{\mathbb{P}(\hat{Y} = 1|S = 1)}$$

$$\text{EO} = k, \text{ if EO} \leq 1; \text{EO} = 1/k, \text{ otherwise,} \quad \text{where } k = \frac{\mathbb{P}(\hat{Y} = 1|S = -1, Y = y)}{\mathbb{P}(\hat{Y} = 1|S = 1, Y = y)},$$

where $y \in \{-1, 1\}$. The EO metric requires that $\hat{Y}$ have equal true and false positive rates between sub-groups. For both metrics, a value closer to 1 indicates fairness.

6.3 Competing Methods

We compare the performance of our algorithm FairSAOML on various interval settings (hyphenated by DI, AGC, and DGC) with six baseline methods. These baselines are chosen from three perspectives: online meta-learning (MaskFTML, FairFML), online fairness learning (FairFML, FairAOGD, FairGLC), and online learning in changing environments (AOD, CBCE).

- **MaskFTML** [6]: the original FTML finds a sequence of meta parameters by simply applying MAML [5] at each round. To focus on fairness learning, this approach is applied to modified datasets by simply removing protected attributes.
- **FairFML** [28] controls bias in an online working paradigm and aims at attaining zero-shot generalization with task-specific adaptation. Different from our FairSAOML, FairFML focuses on a static environment and assumes tasks sampled from an unchangeable distribution.
- **FairAOGD** [11] is proposed for online learning with long-term constraints. In order to fit bias-prevention and compare them to FairSAOML, we specify such constraints as DDP stated in Definition 1.
- **FairGLC** [22] rectifies FairAOGD by square-clipping the constraints in place of $g_i(\cdot)$, $\forall i$.
- **AOD** [23] minimizes the SAR by running multiple online gradient descent algorithms over a set of dense geometric covering intervals.
- **CBCE** [12] adapts to changing environments in an online learning paradigm by combining the sleeping bandits idea with the coin betting algorithm.

---

[1]https://grouplens.org/datasets/movielens/100k/

6.4 Settings

As discussed in Section 5, the performance of our proposed method has been well justified theoretically for machine learning models whose objectives are strongly convex and smooth. However, in machine learning and fairness studies, due to the nonlinearity of neural networks, many problems have a non-convex landscape where theoretical analysis is challenging. Nevertheless, algorithms originally developed for convex optimization problems like gradient descent have shown promising results in practical non-convex settings [6]. Taking inspiration from these successes, we describe practical instantiations for the proposed online algorithm and empirically evaluate the performance in Section 7.

For each task, we set the number of fairness constraints to one, i.e., $m = 1$. For the rest, we follow the same settings as used in online meta-learning [6, 28]. In particular, we meta-train with a support size of 400 for each class and 800 for a query set, whereas 90% (hundreds of datapoints) of task samples for evaluation. Besides, for the *NYSF* dataset, we chose the base of 2, and the total number of experts is 96 for DI, 6 for AGC, and 7 for DGC. Similarly, we chose the base of 3 for the *MovieLens* dataset; hence, the number of experts is 90 for DI, 4 for AGC, and 5 for DGC. All the baseline models used to compare with our proposed approach share the same neural network architecture and parameter settings. All the experiments are repeated ten times with the same settings, and the mean and standard deviation results are reported.

6.5 Implementation Details and Hyperparameter Tuning

Our neural network trained follows the same architecture used in [5], which contains two hidden layers of size 40 with ReLU activation functions. In the training process of the *MovieLens (NYSF)* data, each gradient is computed using a batch size of 200 (800) examples where each binary class contains 100 (400) examples. For each dataset, we tune the following hyperparameters: (1) the initial dual meta parameter λ_0 is chosen from {0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000}; (2) the interval-level gradient steps are chosen from 1 to 10; (3) the number of iterations N_{meta} are chosen from {20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100}; (4) learning rates η_1 and η_2 for updating meta-level parameters in Equations (16) and (15) are chosen from {0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100, 500, 1000}; (5) the positive constant δ used in the augmented term are chosen from {10, 25, 50, 75, 100}.

7 RESULTS

7.1 Overall Performance

The consolidated results, depicted in Figure 4, provide a comprehensive evaluation of the effectiveness and efficiency of the proposed method, utilizing three evaluation metrics: fairness (DP and EO) and model precision (accuracy).

In all the curves presented for various methods, higher values indicate better performance across all plots. The shaded regions in the figures represent standard errors. The results demonstrate that our proposed FairSAOML with all interval settings effectively mitigates bias as the learner encounters more tasks, eventually satisfying the "80%-rule" fairness condition [1], where DP and EO exceed 0.8 in the latter stages. Furthermore, FairSAOML consistently outperforms most alternative approaches in terms of achieving the best model precision, as indicated by the high accuracy scores.

Regarding learning efficiency, our FairSAOML with the DI setting takes the most running time. In contrast, FairSAOML with AGC and DGC settings exhibit the shortest running times when compared to the baseline methods shown in the bar charts of Figure 4. This observation can be attributed to several factors: (1) the number of experts at each time in AGC and DGC significantly
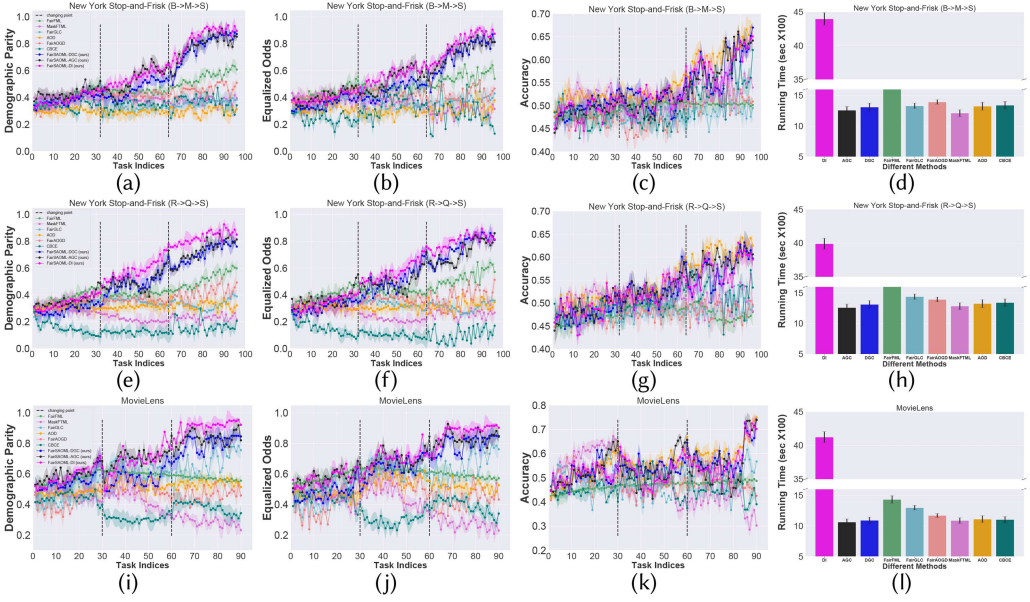
Fig. 4. Model performance over real-world datasets through each time. *NYSF* **(a–c)** B→M→S, **(d–f)** R→Q→S; **(g–i)** *MovieLens*.

decreases compared to the one in DI; (2) only active experts, but not sleeping ones, make contributions for parameter updates; (3) instead of the entire data task, a data subset (support) is used for parameter updates within active experts.

7.2 Adaptability to Changing Environments

The primary objective of our experimental design is to assess the adaptability of FairSAOML concerning fairness and model accuracy as the environment transitions from one to another. To facilitate a clearer visualization of these changing environments, we have manually inserted vertical dotted lines in Figure 4, distinguishing the different environments at specific task indices. Our experimental findings reveal that, while FairSAOML may not initially outperform other baseline methods in the first environment, it excels in adapting to changing conditions. As a result, its performance consistently improves in terms of both model fairness and predictive accuracy as the environment evolves.

In Section 4.2.2, we introduced experts as crucial components in FairSAOML, where the model parameter pair (θ_t, λ_t) at time t is determined by aggregating weighted expert advice. Figure 5 illustrates the evolution of expert weights in FairSAOML-AGC and FairSAOML-DGC. We did not track the weight changes of experts in FairSAOML-DI due to its larger number of experts (96 in *NYSF* and 90 in *MovieLens*). Our observations are as follows: (1) Experts associated with longer intervals receive larger weights, and these weights continue to increase as the learner encounters more tasks; (2) Conversely, experts linked to shorter intervals receive smaller weights and become less influential over time. These findings align with expectations, as assigning heavier weights to experts with longer intervals empowers our FairSAOML to effectively adapt to the volatility in model performance induced by changing environments.

Among the baseline methods, MaskFTML demonstrates superior accuracy performance in the first environment, as evidenced in Figure 4 (c, g, k). However, it falls short when it comes to achieving model fairness, suggesting that merely attempting to obscure the protected attribute
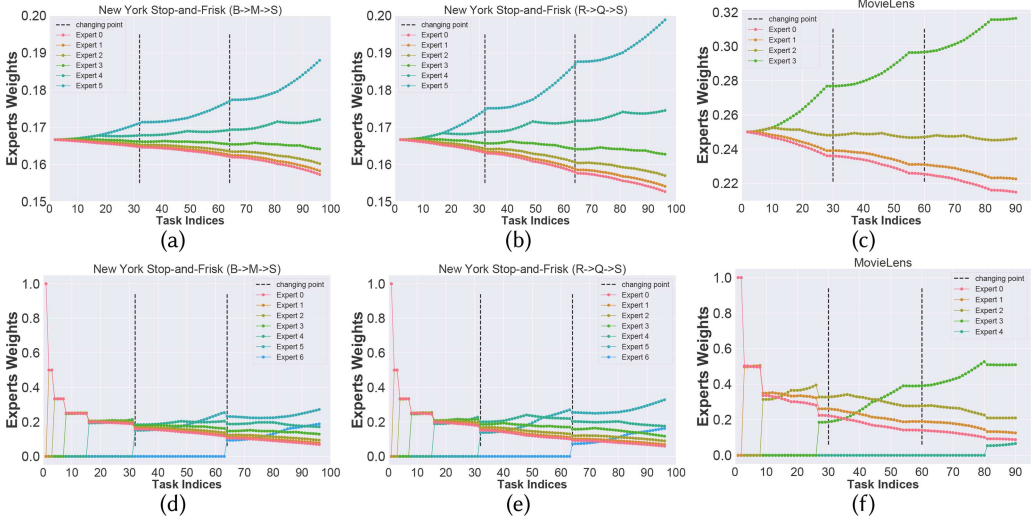
Fig. 5. Expert weight changes over time. **(a–c)** FairSAOML-AGC, **(d–f)** FairSAOML-DGC.

from decision-makers is insufficient to improve prediction fairness. On the other hand, FairFML, FairAOGD, and FairGLC exhibit an ability to mitigate bias in the first environment. Still, they struggle to adapt both fairness and predictive accuracy when the environment undergoes changes. In contrast, AOD and CBCE, originally designed for online learning in dynamic environments, prioritize learning accuracy but do not effectively address model fairness when environmental shifts occur. Furthermore, the pursuit of higher accuracy in AOD often results in a tradeoff in terms of fairness performance. These observations highlight the challenges and tradeoffs involved in achieving a balance between accuracy and fairness across various methods in changing environments.

7.3 Ablation Studies

We conducted ablation studies on the *NYSF* (B→M→S) dataset to assess the contributions of two pivotal components within FairSAOML: expert weights $p_{t,I}$ and the base learner, as described in Section 4.3.1.

To elaborate, meta-level parameters are computed at each time by aggregating expert decisions based on their respective weights. By removing expert weights, all experts contribute equally to the decision-making process. Furthermore, within active experts, base learners, as defined in Equation (10), are employed to update model parameters at an interval level. Without base learners, all active experts share the same model parameters inherited from the previous time and are consequently assigned equal weight. The key insights from the results presented in Figure 6 are as follows: (1) Expert weights play a significant role in FairSAOML, indicating their importance in achieving effective bias control and predictive accuracy; (2) The inclusion of base learners serves to enhance model performance concerning bias control and predictive accuracy. These findings emphasize the critical contributions of expert weights and base learners to the overall effectiveness of the FairSAOML algorithm.

7.4 Sensitive Analysis on Different Bases in AGC and DGC

Sensitive analyses conducted on the *MovieLens* dataset, as depicted in Figure 7, involve the subsetting of intervals using different bases selected from the set $\{2, 3, 4, 5\}$. According to Equations (7) and (9), the configuration with the smallest base value (i.e., 2) results in the highest number of
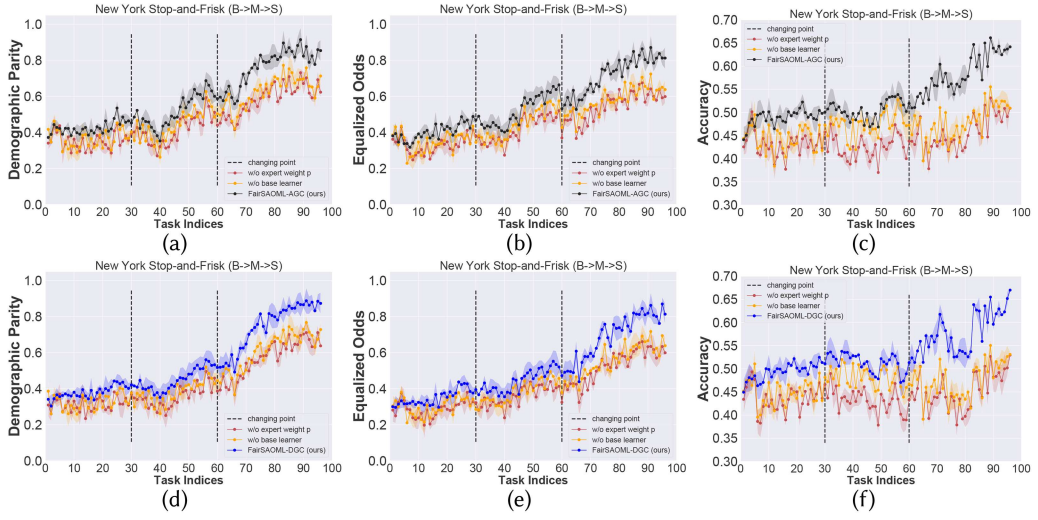
Fig. 6. Performance of ablation studies on the NYSF B→M→S dataset. **(a–c)** FairSAOML-AGC, **(d–f)** FairSAOML-DGC.
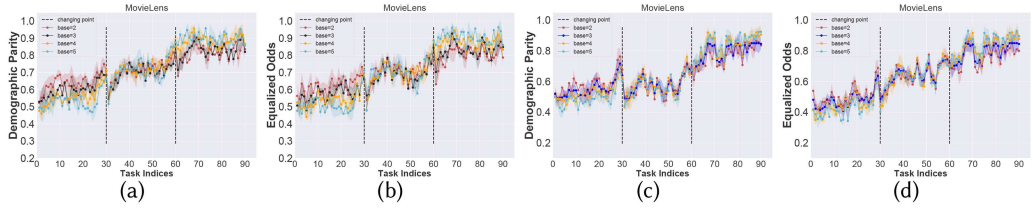

Fig. 7. Different choices of bases for **(a,b)** FairSAOML-AGC and **(c,d)** FairSAOML-DGC on the MovieLens dataset.

experts (6 for AGC and 7 for DGC). Consequently, the largest expert in this setting carries the longest intervals (32 for AGC and 64 for DGC).

Our observations regarding model fairness reveal that settings with smaller bases exhibit slightly better performance than those with larger bases in the first environment. However, the opposite trend is observed in the last environment. This occurs for two main reasons: (1) In the first environment, the largest experts carry more information in the smaller base setting than in the larger base setting; (2) In the last environment, the largest experts in smaller base settings become less pure and incorporate data from different environments, leading to a deterioration in fairness. These findings underscore the sensitivity of the FairSAOML algorithm to the choice of base value and its impact on model fairness, particularly in different environmental contexts.

8 CONCLUSION

To address the challenges of fairness-aware online learning in changing environments, where data tasks are sampled from diverse distributions one after another, we introduce a novel regret measure called FairSAR. FairSAR extends SAR by incorporating long-term fairness constraints. In technical terms, we start by proposing three alternative sets of intervals. At each time step, we dynamically select a target set consisting of multiple intervals from these sets. Next, we introduce a novel learning algorithm, named FairSAOML, to sequentially determine model parameters. In this algorithm, we dynamically activate a subset of experts based on the intervals in the target set and update

their parameters at an interval level. The meta-level model parameters are then obtained by combining the weighted contributions of all experts. Detailed theoretical analysis and accompanying proofs provide justification for the efficiency and effectiveness of our proposed algorithm. We demonstrate upper bounds for loss regret and the violation of fairness constraints. Empirical studies conducted on real-world datasets demonstrate that our method outperforms state-of-the-art online learning techniques in terms of both model accuracy and fairness.

REFERENCES

[1] Dan Biddle. 2006. Adverse impact and test validation: A practitioner's guide to valid and defensible employment testing. Gower, page 220. DOI : https://doi.org/10.4324/9781315263298

[2] Nicolò Cesa-Bianchi and Gábor Lugosi. 2006. *Prediction, Learning, and Games*. Cambridge University Press

[3] Amit Daniely, Alon Gonen, and Shai Shalev-Shwartz. 2015. Strongly adaptive online learning. In *Proceedings of the ICML*.

[4] Cynthia Dwork, et al. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*.

[5] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the ICML*.

[6] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. 2019. Online meta-learning. In *Proceedings of the ICML*.

[7] András Gyorgy, Tamás Linder, and Gábor Lugosi. 2012. Efficient tracking of large classes of experts. *IEEE Transactions on Information Theory* 58, 11 (2012), 6709–6725.

[8] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*. 3323–3331.

[9] Elad Hazan and Edgar Minasyan. 2020. Faster projection-free online learning. In *Proceedings of the 33rd Conference on Learning Theory*.

[10] Elad Hazan, and Comandur Seshadhri. 2007. Adaptive algorithms for online decision problems. *Electronic Colloquium on Computational Complexity (ECCC)*. 14, 088 (2007).

[11] Rodolphe Jenatton, Jim Huang, and Cedric Archambeau. 2016. Adaptive algorithms for online convex optimization with long-term constraints. *Proceedings of the ICML*.

[12] Kwang-Sung Jun, Francesco Orabona, Stephen Wright, and Rebecca Willett. 2017. Improved strongly adaptive online learning using coin betting. In *Proceedings of the AISTATS*.

[13] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara M. Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. 2021. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the ICML*.

[14] Michael Lohaus, Michael Perrot, and Ulrike Von Luxburg. 2020. Too relaxed to be fair. In *Proceedings of the ICML*.

[15] Haipeng Luo and Robert E. Schapire. 2015. Achieving all with no parameters: Adanormalhedge. In *Proceedings of the Conference on Learning Theory*. PMLR, 1286–1304.

[16] Mehrdad Mahdavi, Rong Jin, and Tianbao Yang. 2012. Trading regret for efficiency: online convex optimization with long term constraints. *The Journal of Machine Learning Research* 13.1 (2012), 2503–2528.

[17] Jennifer Miller. 2020. Is an algorithm less racist than a loan officer? Retrieved Sept. 18 2020 from https://www.nytimes.com/2020/09/18/business/digital-mortgages.html

[18] Yuanyu Wan, Bo Xue, and Lijun Zhang. 2021. Projection-free online learning in dynamic environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[19] Zhuoyi Wang, Yuqiao Chen, Chen Zhao, Yu Lin, Xujiang Zhao, Hemeng Tao, Yigong Wang, and Latifur Khan. 2021. CLEAR: Contrastive-prototype learning with drift estimation for resource constrained stream mining. In *Proceedings of the WWW*.

[20] Yongkai Wu, Lu Zhang, and Xintao Wu. 2019. On convexity and bounds of fairness-aware classification. In *Proceedings of the WWW*.

[21] Jiahao Xie, Zebang Shen, Chao Zhang, Boyu Wang, and Hui Qian. 2020. Efficient projection-free online methods with stochastic recursive gradient. In *Proceedings of the AAAI*.

[22] Jianjun Yuan and Andrew Lamperski. 2018. Online convex optimization for cumulative constraints. *Advances in Neural Information Processing Systems*. 6140–6149.

[23] Lijun Zhang, Shiyin Lu, and Tianbao Yang. 2020. Minimizing dynamic regret and adaptive regret simultaneously. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*.

[24] Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. 2018. Adaptive online learning in dynamic environments. *Advances in Neural Information Processing Systems*. 1323–1333.

[25] Chen Zhao. 2021. *Fairness-Aware Multi-Task and Meta Learning*. Ph. D. Dissertation.

[26] Chen Zhao and Feng Chen. 2019. Rank-based multi-task learning for fair regression. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*.

[27] Chen Zhao and Feng Chen. 2020. Unfairness discovery and prevention for few-shot regression. In *Proceedings of the 2020 IEEE International Conference on Knowledge Graph (ICKG)*.

[28] Chen Zhao, Feng Chen, and Bhavani Thuraisingham. 2021. Fairness-aware online meta-learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*.

[29] Chen Zhao, Feng Mi, Xintao Wu, Kai Jiang, Latifur Khan, and Feng Chen. 2022. Adaptive fairness-aware online meta-learning for changing environments. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2565–2575.

[30] Martin Zinkevich. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the ICML*.