# Unbiased Subdata Selection for Fair Classification: A Unified Framework and Scalable Algorithms

**Qing Ye**                                                                      YQING1@VT.EDU
*Department of Industrial and Systems Engineering*
*Virginia Tech*
*Blacksburg, VA 24061, USA*

**Weijun Xie**                                                                      WXIE@VT.EDU
*Department of Industrial and Systems Engineering*
*Virginia Tech*
*Blacksburg, VA 24061, USA*

**Editor:**

## Abstract

As an important problem in modern data analytics, classification has witnessed varieties of applications from different domains. Different from the conventional classification approaches, fair classification concerns the issues of unintentional biases against the sensitive features (e.g., gender, race). Due to high nonconvexity of fairness measures, existing methods are often unable to model exact fairness, which can cause inferior fair classification outcomes. This paper fills the gap by developing a novel unified framework to jointly optimize accuracy and fairness. The proposed framework is versatile and can incorporate different fairness measures studied in literature precisely as well as can be applicable to many classifiers including deep classification models. Specifically, in this paper, we first prove Fisher consistency of the proposed framework. We then show that many classification models within this framework can be recast as mixed-integer convex programs, which can be solved effectively by off-the-shelf solvers when the instance sizes are moderate and can be used as benchmarks to compare the efficiency of approximation algorithms. We prove that in the proposed framework, when the classification outcomes are known, the resulting problem, termed "unbiased subdata selection, is strongly polynomial-solvable and can be used to enhance the classification fairness by selecting more representative data points. This motivates us to develop an iterative refining strategy (IRS) to solve the large-scale instances, where we improve the classification accuracy and conduct the unbiased subdata selection in an alternating fashion. We study the convergence property of IRS and derive its approximation bound. More broadly, this framework can be further leveraged to improve classification models with unbalanced data by taking $F_1$ score into consideration. Finally, we numerically demonstrate that the proposed framework can consistently yield better fair classification outcomes than existing methods.

**Keywords:** Fair Classification, Subdata Selection, Mixed-Integer Program, Approximation Algorithms, CNN, Unbalanced Data

## 1. Introduction

As an important problem in modern data analytics, classification has witnessed varieties of applications including diagnosis (De Bruijne, 2016), face recognition (Naseem et al., 2010),

text categorization (Zhang and Oles, 2001), and microarray gene expression (Pirooznia et al., 2008). Different from the conventional classification approaches, fair classification concerns the discrimination against the protected groups when performing the classification tasks. Recently, fairness in machine learning has attracted much attention since more and more evidences have shown that the traditional methods might cause biases against sensitive features, such as gender, race, or ethnicity. One example of machine learning biases is in criminal risk assessment using the COMPAS recidivism algorithm (Jeff Larson and Angwin). In 2016, Jeff Larson and Angwin analyzed the COMPAS recidivism algorithm and concluded that black defendants were twice as likely as white defendants to be misclassified as being at a high risk of recidivism. To prevent such discrimination, many notions of machine learning fairness, in particular, classification fairness, have been established. For example, one popular fair classification measure is overall misclassification rate (OMR) (Zafar et al., 2019), which accounts for the misclassification rates of disparate mistreatment among different groups; another fair classification measure is demographic parity (Menon and Williamson, 2018), which measures statistical independence between the classification outcomes and the sensitive feature. Due to nonconvexity, existing methods are often unable to model exact fairness, which can cause inferior fair classification outcomes. To foster the impact of machine learning on society progressively, this paper paves a generic way to incorporate the exact fairness measures to improve classification fairness and computational efficiency.

## 1.1 Relevant Literature

In this subsection, we present an overview of fairness measures and existing methods for fair classification.

**Fairness Measures:** According to Verma and Rubin (2018), fairness measures for classification problems can be classified as statistical measures, similarity-based measures, and casual reasoning. Statistical measures such as group fairness, predictive parity, and test-fairness are developed based on confusion matrix with their different emphases on actual outcome, predicted outcome, or predicted probabilities. Similarity-based measures such as causal discrimination and fairness through unawareness incorporate insensitive features using data similarity. Casual reasoning such as counter-factual fairness and no unresolved discrimination define fairness measures for causal graphs that are used to build fair classifiers (Kilbertus et al., 2017). This paper mainly focuses on the fundamental statistical measures, where disparate treatment, disparate impact, and disparate mistreatment are the common fairness notions (Zafar et al., 2019; Taskesen et al., 2020). Specifically, a classifier achieves no disparate treatment if the prediction results are independent of the sensitive feature, while a classifier does not suffer from the disparate impact if its proportional prediction outcome of a specific label is the same to the different groups of the sensitive feature. This paper first studies one popular no disparate mistreatment fairness measure (i.e., OMR fairness) that requires the same misclassification rates for the different groups. We extend the results to two other no disparate mistreatment fairness measures–false positive rate and equal opportunity, and a popular no disparate impact fairness measure–demographic parity.

**Fair Classification:** Fair classification can be achieved through pre-processing (Hajian and Domingo-Ferrer, 2012; Calmon et al., 2017; Kamiran and Calders, 2012), in-processing

| Existing Works | SVM | Kernel SVM | Multiclass SVM | Logistic Regression | Black-box Classifiers (e.g., CNN) |
|---|---|---|---|---|---|
| Donini et al. (2018), etc. | ✓ | ✓ | | | |
| Zafar et al. (2019), Olfat and Aswani (2017), Zafar et al. (2017), etc. | ✓ | ✓ | | ✓ | |
| Goel et al. (2018), Kamishima et al. (2012), Menon and Williamson (2018), Agarwal et al. (2018), Taskesen et al. (2020), etc. | | | | ✓ | |
| This paper | ✓ | ✓ | ✓ | ✓ | ✓ |

Table 1: Summary of Recent Studies on Fair Classification

(Kamishima et al., 2012; Agarwal et al., 2018; Zafar et al., 2017), or post-processing (Hardt et al., 2016; Pleiss et al., 2017; Fish et al., 2016) approaches. In this paper, we focus on the most flexible method (i.e., in-processing approach) to generate fairness-aware classifiers. To reduce discrimination in machine learning, many recent studies incorporate fairness measures as constraints (Donini et al., 2018; Menon and Williamson, 2018) or as a part of the objective function (Balashankar et al., 2019; Aghaei et al., 2019). For instance, fairness constraints of demographic parity (Menon and Williamson, 2018; Goel et al., 2018; Olfat and Aswani, 2017) or equal opportunity (Hardt et al., 2016; Zafar et al., 2017; Menon and Williamson, 2018) were added to the model formulations, while the fairness measures were suggested by (Balashankar et al., 2019) to be added into the objective function as a regularization term. Due to nonconvexity of the most fairness measures, the fairness-related regularization or constraints usually were approximated to be convex ones (Olfat and Aswani, 2017; Agarwal et al., 2018), which can be then reduced as efficiently solvable convex programs. These approximations are often inexact and can result in less fair results. Different from existing ones, this paper enforces the exact fairness by building mixed-integer programs, which enable to compute the fairness measures precisely. When incorporating fairness measures to construct fairness-aware classifiers, most of the existing works were only designated for a specific classifier. Table 1 summarizes recent studies. As far as we are concerned, there is no existing approaches that are applicable to multiclass classification or black-box classifiers under fairness. The framework we proposed is flexible and can be extended to different types of fairness measures or different classifiers. It is also applicable to black-box classifiers. For example, it can be adapted to improve the fairness of popular deep classification models. Notably, our approach also shows advantages in the numerical study.

| Model | $\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})$ | $\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u})$ |
|:---:|:---:|:---:|
| SVM (12) | $1/N \sum_{i \in [N]} z_i(u_i - t)$ | $\|\boldsymbol{w}\|_2^2$ |
| Multiclass SVM (27) | $1/N \sum_{i \in [N]} \sum_{j \in [K], j \neq y_i} (1 - z_{ij})(u_{ij} - t)$ | $\sum_{j \in [K]} \|\boldsymbol{w}_j\|_2^2$ |
| Logistic Regression (31) | $1/N \sum_{i \in [N]} z_i (t - u_i)$ | $\|\boldsymbol{w}\|_2^2$ |
| Deep Learning (33) | $1/N \sum_{i \in [N]} z_i (t - u_i)$ | $0$ |

Table 2: Examples of Different Models under the Unified Framework (1)

## 1.2 Summary of Contributions

This paper first studies fair support vector machine (SVM) as a motivating example. We formulate the generalized SVM (GSVM) model as a mixed-integer program to model exact fairness, and then study the model properties, develop scalable algorithms for solving the generalized SVM with fairness (GSVMF) as well as study the extensions to other popular classification models including deep learning with different fairness measures. Specifically, our framework can be formulated as

$$
\min_{(\boldsymbol{w}, b, \boldsymbol{u}) \in U, \boldsymbol{z}} \left\{ \underbrace{\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})}_{\text{Classification Outcome}} + \lambda \underbrace{\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u})}_{\text{Regularization}} + \rho \underbrace{\mathcal{F}(\boldsymbol{z})}_{\text{Fairness Measure}} : \boldsymbol{z} \in Z \right\}, \tag{1}
$$

where $t \in \mathbb{R}$ is the prediction threshold, vector $\boldsymbol{u}$ represents classification outcomes (e.g., violation margins, prediction probabilities) under a particular classifier specified by unknown parameters $\boldsymbol{w}, b$, $\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u})$ denotes the regularization term for the classifier with tuning parameter $\lambda \geq 0$, binary vector $\boldsymbol{z} \in Z$ indicates the correct classification corresponding to subdata selection decisions, and binary set $Z$ denotes the domain of $\boldsymbol{z}$. The binary vector $\boldsymbol{z}$ can be used to model the statistical fairness measures precisely without any approximation. In the unified framework (1), function $\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})$, which is often bilinear, denotes the classification outcomes and the function $\mathcal{F}(\boldsymbol{z})$ denotes a fairness measure, where a Pareto optimality between accuracy and fairness is achieved by a proper penalty parameter $\rho > 0$. Table 2 displays the function $\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})$ and function $\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u})$ for well-known classifiers. Table 3 shows function $\mathcal{F}(\boldsymbol{z})$ for four popular fairness measures. Besides, since the optimized binary decisions $\boldsymbol{z}$ take values $\{0, 1\}$ according to the tradeoff of accuracy and fairness, the proposed method has the subdata selection interpretation. That is, we will select the data points such that their $z$ values are equal to one and train the classifier only using the selected subdata. Therefore, the proposed framework can select unbiased data points for fair classification and can be naturally solved using the iterative refining strategy with two subroutines, i.e., subdata selection and classification using the selected subdata. We illustrate the subdata selection in Figure 1. Note that Figure 1a demonstrates the decision boundary for the vanilla classifier which may not be fair to the protected groups, while Figure 1b shows that the unbiased subdata selection in the proposed framework can reduce the unfairness score to 0% while achieving the same accuracy as vanilla classifier.

As illustrated in Figure 2, the main contributions of this paper are summarized below:

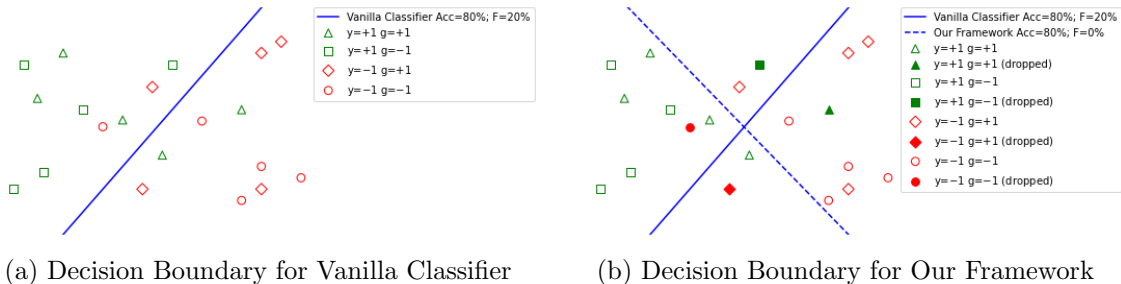| Fairness Measures | $\mathcal{F}(\boldsymbol{z})$ |
|---|---|
| Overall Misclassification Rate (11a) | $\left| \sum_{i \in \mathcal{D}_+} z_i / D_+ - \sum_{i \in \mathcal{D}_-} z_i / D_- \right|$ |
| False Positive Rate (21a) | $\left| \sum_{i \in \mathcal{D}_{+-}} z_i / D_{+-} - \sum_{i \in \mathcal{D}_{--}} z_i / D_{--} \right|$ |
| Equal Opportunity (23a) | $\left| \sum_{i \in \mathcal{D}_{++}} z_i / D_{++} - \sum_{i \in \mathcal{D}_{-+}} z_i / D_{-+} \right|$ |
| Demographic Parity (25) | $\left| \sum_{i \in \mathcal{D}_{++}} z_i / D_+ + \sum_{i \in \mathcal{D}_{+-}} (1 - z_i) / D_+ \right.$ $\left. - \sum_{i \in \mathcal{D}_{-+}} z_i / D_- - \sum_{i \in \mathcal{D}_{--}} (1 - z_i) / D_- \right|$ |

Table 3: Examples of Different Fairness Measures Expressed as $\mathcal{F}(\boldsymbol{z})$



(a) Decision Boundary for Vanilla Classifier      (b) Decision Boundary for Our Framework

Figure 1: The Role of Unbiased Subdata Selection: A Comparison of Vanilla Classifier and Our Framework. Note that $y \in \{-1, 1\}$ denotes the class labels, $g \in \{-1, 1\}$ denotes the protected groups, $Acc$ denotes the classification accuracy, and $F$ denotes the overall misclassification rate (OMR) (lower $F$ is better).

(i) We propose a generalized SVM (GSVM) model for the binary classification problems and prove that it is Fisher consistent. We incorporate the overall misclassification rate (OMR) precisely in the generalized SVM model, termed "GSVMF";

(ii) We propose an exact mixed-integer conic programming formulation for GSVMF, which can be solved by off-the-shelf solvers for the moderate-sized instances;

(iii) For the large-scale instances, we propose to solve GSVMF effectively to near-optimality using the iterative refining strategy (IRS) by employing an unbiased subdata selection when fixing the classification outcomes and executing a classifier given the selected subdata in an alternative fashion. In the proposed IRS, the subdata selection, albeit resorting to a binary program, is strongly polynomial-solvable with time complexity of $O(N \log N)$, where $N$ denotes the number of data points. We also study the convergence property of IRS and derive its approximation bound;

(iv) We show that the proposed GSVMF is also amenable to represent other fairness measures such as false positive rate, equal opportunity, and demographic parity; and

(v) We further present the extensions to fair multiclass classification, logistic regression with fairness, kernel SVM with fairness, fair deep learning, and classification with unbalanced data.

The remainder of this paper is organized as follows. Section 2 presents the model formulation and model properties of GSVMF. Section 3 studies the unbiased subdata selection method and the iterative refining strategy for solving GSVMF. Section 4 and Section 5 demonstrate the extensions to other fairness measures and different classifiers, respectively. Section 6 presents the application to classification with unbalanced data. Section 7 shows the numerical experiments. Section 8 concludes the paper.
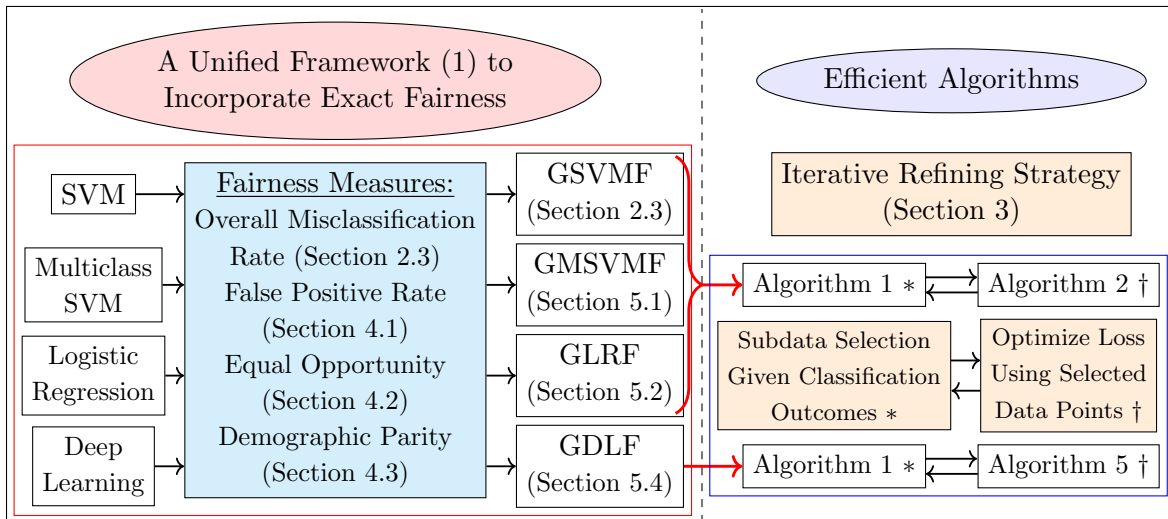


Figure 2: The Roadmap of Our Contributions in This Paper. Note that Algorithm 1 is subdata selection procedure, and Algorithm 2 and Algorithm 5 employ the classifiers on the given the selected subdata.

*Notation:* The following notation is used throughout the paper. We use bold-letters (e.g., $\boldsymbol{x}$, $\boldsymbol{u}$) to denote vectors or matrices, and use corresponding non-bold letters to denote their components. Given a real number $t$, we let $\lceil t \rceil$ be its round-up and $\lfloor t \rfloor$ be its round-down. Given an integer $n$, we let $[n] := \{1, 2, \ldots, n\}$. The indicator function $\mathbb{I}(\boldsymbol{x} \in R) = 1$ if $\boldsymbol{x} \in R$, and 0, otherwise. Given a vector $\boldsymbol{x} \in \mathbb{R}^n$, we let $\text{Supp}(\boldsymbol{x})$ denote its support, i.e., $\text{Supp}(\boldsymbol{x}) := \{i : x_i \neq 0\}$. Given two subsets $S, T \subseteq [n]$, we let $S \Delta T$ denote the symmetric difference of sets $S$ and $T$, i.e., $S \Delta T := (S \setminus T) \cup (T \setminus S)$. Additional notation will be introduced as needed.

## 2. A Motivating Special Case: Generalized Support Vector Machine with OMR Fairness

In this section, we will present model formulations and analyze properties for generalized support vector machine (GSVM) and extend GSVM with OMR fairness measure. Results of this section can be generalized to those having a different classifier or a distinct fairness measure.

Support vector machine (SVM) is a traditional approach for binary classification problems (Suykens and Vandewalle, 1999). The aim of SVM is to construct a classifier to predict labels, i.e., the predictor $\tilde{\boldsymbol{x}} \in \mathbb{R}^n$ is used to predict the label $\tilde{y} \in \{-1, 1\}$. Given a dataset

$\{(\boldsymbol{x}_i, y_i)\}_{i \in [N]} \subseteq \mathbb{R}^n \times \{-1, 1\}$ with $N$ data points, SVM admits the following form

$$\min_{\boldsymbol{w}, b, \boldsymbol{u}} \left\{ \sum_{i \in [N]} u_i + \lambda \|\boldsymbol{w}\|_2^2 : y_i(\boldsymbol{x}_i^\top \boldsymbol{w} + b) \geq 1 - u_i, u_i \geq 0, \forall i \in [N] \right\}, \tag{2}$$

where $\lambda \in \mathbb{R}$ is a tuning parameter, and $\boldsymbol{w} \in \mathbb{R}^n$, $b \in \mathbb{R}$ are unknown parameters of the classifier.

## 2.1 Generalized SVM (GSVM) and Mixed-Integer Programming Formulation

Note that in SVM (2), for each data point $i \in [N]$, the violation margin $u_i$ helps predict the classification results, i.e., if $u_i$ is less than or equal to some threshold, then we predict $i$th data point to be of the same label as $y_i$, and misclassify it, otherwise. Thus, this observation inspires us to improve SVM using an indicator function $\mathbb{1}(u_i \leq t)$ to minimize the sum of misclassification margins, where t is the threshold. Specifically, we propose the following generalized SVM (GSVM)

$$v := \min_{\boldsymbol{w}, b, \boldsymbol{u}} \left\{ \frac{1}{N} \sum_{i \in [N]} (u_i - t) \mathbb{1}(u_i \leq t) + \lambda \|\boldsymbol{w}\|_2^2 : y_i(\boldsymbol{x}_i^\top \boldsymbol{w} + b) \geq 1 - u_i, u_i \geq 0, \forall i \in [N] \right\}, \tag{3}$$

We remark that (i) if $t \to +\infty$, GSVM (3) becomes SVM (2); (ii) one can choose $t$ properly through cross-validation; and (iii) although being intractable, GSVM (3) can be useful to incorporate many fairness measures and we will show later that it can be solved effectively via an off-the-shelf solver for moderate-sized instances and an iterative refining strategy for large-scale instances.

Next, we observe that in GSVM (3), for each $i \in [N]$, the indicator function $\mathbb{1}(u_i \leq t)$ can be replaced by the binary variable $z_i$, where $z_i = 1$ indicates correct classification and $z_i = 0$ indicates misclassification with respect to the threshold $t$. We see that, GSVM (3) admits an equivalent mixed-integer biconvex programming formulation.

**Proposition 1** *GSVM is equivalent to*

$$v := \min_{\boldsymbol{w}, b, \boldsymbol{u}, \boldsymbol{z}} \frac{1}{N} \sum_{i \in [N]} z_i(u_i - t) + \lambda \|\boldsymbol{w}\|_2^2, \tag{4a}$$

$$\text{(GSVM)} \qquad s.t. \quad y_i(\boldsymbol{x}_i^\top \boldsymbol{w} + b) \geq 1 - u_i, \forall i \in [N], \tag{4b}$$

$$u_i \geq 0, \forall i \in [N], \tag{4c}$$

$$z_i \in \{0, 1\}, \forall i \in [N], \tag{4d}$$

**Proof** The proof follows by observing that at optimality, for each $i \in [N]$, we must have $z_i = 1$ if $u_i \leq t$, and 0, otherwise; that is, $z_i = \mathbb{1}(u_i \leq t)$. ∎

The binary variables $\boldsymbol{z}$ enable us to distinguish the correct classified and misclassified data points. Thus, we can model the fairness using these binary variables. Note that the biconvex program (4) can be equivalently reformulated as a mixed-integer conic program (MICP) with the following McCormick inequalities to linearize the bilinear terms $\{z_i u_i\}_{i \in [N]}$

$$s_i \leq M_u z_i, \quad s_i \leq u_i, \quad s_i \geq 0, \quad s_i \geq u_i - M_u(1 - z_i), \forall i \in [N] \tag{5}$$

where $M_u$ is a positive large number and its derivation will be postponed to the next proposition. The result of MICP is summarized below.

**Proposition 2** *GSVM can be formulated as the following MICP*

$$v := \min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z},\boldsymbol{s}} \left\{ \frac{1}{N} \sum_{i \in [N]} (s_i - z_i t) + \lambda \|\boldsymbol{w}\|_2^2 : (4b) - (4d), (5) \right\}. \tag{6}$$

This result allows us to use off-the-shelf solvers such as Gurobi and CPLEX to solve (6) to optimality when the number of data points is not very large, and we can also use the optimal solution from this exact formulation to test the effectiveness of the proposed iterative refining strategy.

The next proposition shows how to compute the big-M coefficient $M_u$ from GSVM (6) in closed-form. The proof idea is to (i) use the optimality condition; and (ii) discuss different cases of label $y_i$ for each $i \in [N]$.

**Proposition 3** *Suppose that $L_1 = \sqrt{t\lambda^{-1}}$ and $L_2 = \max_{i \in [N]} \|\boldsymbol{x}_i\|_2$. Then $M_u$ can be chosen as $M_u = 2 + 2L_1 L_2$.*

**Proof** First of all, observe that the optimal value of GSVM (4) must satisfy $v \leq 0$ since $\boldsymbol{w} = \boldsymbol{0}$, $b = 0$, $\boldsymbol{u} = t$, $\boldsymbol{z} = 0$ is a feasible solution with objective value equal to 0. Thus, there exists an optimal solution $(\boldsymbol{w}^*, b^*, \boldsymbol{u}^*, \boldsymbol{z}^*)$ such that

$$\frac{1}{N} \sum_{i \in [N]} z_i^*(u_i^* - t) + \lambda \|\boldsymbol{w}^*\|_2^2 \leq 0.$$

Since $\sum_{i \in [N]} z_i^*(u_i^* - t) \geq -Nt$, thus $\lambda \|\boldsymbol{w}^*\|_2^2 \leq t$. Therefore, we can upper bound $\|\boldsymbol{w}^*\|_2$ as $\|\boldsymbol{w}^*\|_2 \leq \sqrt{t\lambda^{-1}} = L_1$. Next, we are going to derive the bound $M_b$ for $|b^*|$. Now letting $a_i := -y_i(\boldsymbol{x}_i^\top \boldsymbol{w} + b) + 1 - t$, we have $u_i = \max\{a_i + t, 0\}$ for each $i \in [N]$. Projecting out variables $\boldsymbol{u}$ and using the fact that $\max\{a_i + t, 0\} \leq t$ is equivalent to $a_i \leq 0$ since $t \geq 0$, GSVM (3) is equivalent to

$$v := \min_{\boldsymbol{w},b} \left\{ \frac{1}{N} \sum_{i \in [N]} \max\{a_i := -y_i(\boldsymbol{x}_i^\top \boldsymbol{w} + b) + 1 - t, -t\} \mathbb{1}(a_i \leq 0) + \lambda \|\boldsymbol{w}\|_2^2 \right\}.$$

Let us denote $a_i^* = -y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^* + b^*) + 1 - t$. We find $M_b$ by discussing the following four cases for each $i \in [N]$:

Case i. If $y_i = +1$ and $a_i^* \leq -t$, then we must have $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^* + b^*) \geq 1$. According to Cauchy-Schwartz inequality $|\boldsymbol{x}_i^\top \boldsymbol{w}| \leq L_1 L_2$ and triangle inequality, this inequality (i.e., $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^* + b^*) \geq 1$) will be guaranteed by choosing $M_b \geq 1 + L_1 L_2$.

Case ii. If $y_i = +1$ and $a_i^* \geq -t$, then we must have $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^* + b^*) \leq 1$. It is sufficient to choose $M_b \geq 1 + L_1 L_2$ to guarantee the inequality.

Case iii. If $y_i = -1$ and $a_i^* \leq -t$, then we must have $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^* + b^*) \geq 1$. It is sufficient to choose $M_b \geq 1 + L_1 L_2$ to guarantee the inequality.

Case iv. If $y_i = -1$ and $a_i^* \geq -t$, then we must have $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^* + b^*) \leq 1$. It is sufficient to choose $M_b \geq 1 + L_1 L_2$ to guarantee the inequality.

Finally, to ensure that the constraint (4b) holds for each $i \in [N]$, according to Cauchy-Schwartz inequality $|\boldsymbol{x}_i^\top \boldsymbol{w}| \leq L_1 L_2$, it is sufficient to choose

$$M_u \geq |1 - y_i(\boldsymbol{x}_i^\top \boldsymbol{w} + b)| \quad \Longleftarrow \quad M_u \geq 1 + M_b + L_1 L_2 \geq 1 + |b^*| + L_1 L_2.$$

Therefore, $M_u = 1 + M_b + L_1 L_2 = 2 + 2L_1 L_2$ suffices. ∎

## 2.2 Fisher Consistency of GSVM

In this subsection, we prove Fisher consistency of the proposed GSVM (4), which further demonstrates the correctness of the proposed model. Given a realization $\boldsymbol{x}$ of random observation $\tilde{\boldsymbol{x}}$, let us define

$$p(\boldsymbol{x}) = \mathbb{P}\left\{\tilde{y} = 1 \mid \tilde{\boldsymbol{x}} = \boldsymbol{x}\right\},$$

i.e., $p(\boldsymbol{x})$ defines the probability that the label of data $\boldsymbol{x}$ equals to 1. Following Lee et al. (2004); Lin (2002); Liu (2007), we consider the Bayesian decision rule $\phi_B(\boldsymbol{x})$, which admits the following form

$$\phi_B(\boldsymbol{x}) = \begin{cases} 1, \text{if } p(\boldsymbol{x}) \geq 0.5, \\ -1, \text{if } p(\boldsymbol{x}) < 0.5, \end{cases} \tag{7}$$

i.e., we assign a label for the realization $\boldsymbol{x}$ according to whether the probability $p(\boldsymbol{x})$ is greater than 0.5 or not. In formulation (4), we will use a more general classifier $f(\boldsymbol{x})$ instead of the linear one $\boldsymbol{x}^\top \boldsymbol{w} + b$. Conditioning on the realization $\tilde{\boldsymbol{x}} = \boldsymbol{x}$ in formulation (4), we are trying to find a separator $f(\boldsymbol{x})$ to minimize the conditional expected surrogate loss. Given a realization $\boldsymbol{x}$ of the random data, we suppose that the classifier $f(\boldsymbol{x})$ is normalized to be between $-t - 1$ and $t + 1$. The classification problem is equivalent to

$$\min_{f(\boldsymbol{x}), \tilde{u}} \left\{\mathbb{E}[(\tilde{u} - t)\mathbb{1}\,(\tilde{u} \leq t)|\tilde{\boldsymbol{x}} = \boldsymbol{x}] : \tilde{y}f(\tilde{\boldsymbol{x}}) \geq 1 - \tilde{u}, \tilde{u} \geq 0, -t - 1 \leq f(\boldsymbol{x}) \leq t + 1\right\}. \tag{8}$$

Projecting out random variable $\tilde{u}$, formulation (8) is equivalent to

$$\min_{f(\boldsymbol{x})} \left\{L(f(\boldsymbol{x})) := \mathbb{E}_{\tilde{y}}\left[\max\{-\tilde{y}f(\tilde{\boldsymbol{x}}) + 1 - t, -t\}\mathbb{1}(\tilde{y}f(\tilde{\boldsymbol{x}}) \geq t - 1)|\tilde{\boldsymbol{x}} = \boldsymbol{x}\right] : \right.$$
$$\left. -t - 1 \leq f(\boldsymbol{x}) \leq t + 1\right\}. \tag{9}$$

Thus, the expected loss minimization problem can be formally defined as

$$f^*(\boldsymbol{x}) \in \arg\min_{f(\boldsymbol{x})} \left\{L(f(\boldsymbol{x})) : -t - 1 \leq f(\boldsymbol{x}) \leq t + 1\right\}, \tag{10}$$

where we let $f^*(\boldsymbol{x})$ denote an optimal classifier.

Next, following the spirit of seminal works (Lee et al., 2004; Lin, 2002; Liu, 2007), we make a formal definition of Fisher consistency for the binary classification problem.

**Definition 4** (**Fisher Consistency**) *For any realization $\boldsymbol{x}$ of random data $\tilde{\boldsymbol{x}}$, its loss function $L(\boldsymbol{f}(\boldsymbol{x}))$ is Fisher consistent if*

$$\phi_B(\boldsymbol{x}) = 2\mathbb{I}(p(\boldsymbol{x}) \geq 0.5) - 1 = 2\mathbb{I}(\boldsymbol{f}^*(\boldsymbol{x}) \geq t - 1) - 1,$$

*where $\boldsymbol{f}^*(\boldsymbol{x})$ is defined in (10).*

Now we are ready to show the consistency of the loss function $L(\boldsymbol{f}(\boldsymbol{x}))$.

**Proposition 5** *Given $t > 0$, the loss function $L(\boldsymbol{f}(\boldsymbol{x}))$ defined in (9) is Fisher consistent.*

**Proof** Note that according to (9), we have

$$\begin{aligned}
L(f(\boldsymbol{x})) &= p(\boldsymbol{x}) \max\{-f(\boldsymbol{x}) + 1 - t, -t\}\mathbb{1}(f(\boldsymbol{x}) \geq t - 1) \\
&\quad + (1 - p(\boldsymbol{x}))\max\{f(\boldsymbol{x}) + 1 - t, -t\}\mathbb{1}(f(\boldsymbol{x}) \leq 1 - t).
\end{aligned}$$

Thus, minimizing the loss function in (10) is equivalent to solving the following minimization problem

$$\min_{f(\boldsymbol{x})}\left\{L(f(\boldsymbol{x})) : -t - 1 \leq f(\boldsymbol{x}) \leq t + 1\right\}$$

where a minimizer is $f^*(\boldsymbol{x}) = t + 1$ if $p(\boldsymbol{x}) \geq 0.5$, and $-t - 1$, otherwise. Thus, we have

$$\phi_B(\boldsymbol{x}) = 2\mathbb{I}(p(\boldsymbol{x}) \geq 0.5) - 1 = 2\mathbb{I}(\boldsymbol{f}^*(\boldsymbol{x}) \geq t - 1) - 1.$$

∎

Theorem 5 shows that similar to SVM (Lin, 2002), GSVM (4) is also Fisher consistent. This demonstrates that the performance of the proposed GSVM (4) is at least as good as SVM for the binary classification problems. We remark that the similar Fisher consistency result holds for the multi-class SVM.

### 2.3 GSVM with Fairness (GSVMF)

In this subsection, we will illustrate the modeling power of GSVM (4) by augmenting it with one popular fairness measure, overall misclassification rate (OMR). Extensions to other fairness measures can be found in Section 4.

The OMR (see, e.g., Zafar et al. 2019) fairness accounts for the misclassification measure of disparate mistreatment. The classifier has no disparate mistreatment if different groups have the same misclassification rates. Suppose that there is a protected feature corresponding to two groups denoted by $g \in \{-1, 1\}$. Then OMR fairness measure can be defined as below.

**Definition 6** *Given a threshold $t > 0$, the OMR fairness measure for GSVM (3) is*

$$F_{\mathrm{OMR}} = \left|\mathbb{P}(\tilde{u} \leq t|\tilde{g} = +1) - \mathbb{P}(\tilde{u} \leq t|\tilde{g} = -1)\right|.$$

The fairness measure function $F_{\text{OMR}}$ is the absolute difference of the misclassification rates of both groups. We then use the binary variables $\boldsymbol{z}$ to exactly represent the OMR fairness measure

$$F_{\text{OMR}} = |\mathbb{P}(\tilde{u} \leq t|\tilde{g} = +1) - \mathbb{P}(\tilde{u} \leq t|\tilde{g} = -1)| = \left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right|, \tag{11a}$$

where two sets

$$\mathcal{D}_+ = \{i \in [N] : g_i = +1\}, \mathcal{D}_- = \{i \in [N] : g_i = -1\} \tag{11b}$$

represent index subsets of data points from each protected group, respectively, and $D_+ = |\mathcal{D}_+|$ is the number of group $g = +1$ data points, and $D_- = |\mathcal{D}_-|$ is the number of group $g = -1$ data points.

To prevent discrimination in prediction, we penalize the unfairness in our proposed GSVM (4) with nonnegative penalty $\rho$. Therefore, we formulate the following GSVMF model

$$v_\rho := \min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \left\{ \underbrace{\frac{1}{N} \sum_{i \in [N]} z_i(u_i - t)}_{\mathcal{A}(\boldsymbol{z},t\boldsymbol{e}-\boldsymbol{u})} + \lambda \underbrace{\|\boldsymbol{w}\|_2^2}_{\mathcal{R}(\boldsymbol{w},b,\boldsymbol{u})} + \rho \underbrace{\left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right|}_{\mathcal{F}(\boldsymbol{z})} : (4b) - (4d) \right\} \tag{12}$$

where $\rho$ is the penalty factor that balances the prediction accuracy and fairness. Clearly, the proposed GSVMF (12) is a special case of the framework (1). Note that GSVMF (12) has non-convex terms $\{z_i u_i\}_{i \in [N]}$. We can linearize them using McCormick inequalities (5). Thus, similar to Theorem 2, GSVMF (12) can be formulated as a MICP.

**Proposition 7** *Formulation GSVMF is equivalent to*

$$v_\rho := \min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z},\boldsymbol{s}} \left\{ \frac{1}{N} \sum_{i \in [N]} (s_i - z_i t) + \lambda \|\boldsymbol{w}\|_2^2 + \rho \left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right| : (4b) - (4d), (5) \right\}.$$

We remark that for the moderate-sized instances, one might be able to solve both MICPs easily via off-the-shelf solvers such as Gurobi and CPLEX.

The observation that GSVMF (12) is a special case of the framework (1) leads to the following remarks.

(i) The binary variables $\boldsymbol{z}$ indicate the subdata selection decisions. Data points with their corresponding $z-$variables taking value of 1 are the potential unbiased ones selected by the model, and the other data points are ignored. This gives rise to an iterative refining strategy described in Section 3 that improves the subdata selection decisions $\boldsymbol{z}$ and classification accuracy in an alternating way;

(ii) The proposed GSVMF (12) can be quite flexible. Indeed, if we replace $u_i$ by the probabilities of classification outcomes and let $t$ denote the misclassification threshold, then GSVMF (12) can be generalized to other classification models with fairness including logistic regression and convolutional neural networks (CNNs). These facts will be elaborated in the subsequent sections;

(iii) Since the classification results are discrete (e.g., $\{0, 1\}$ for the binary classification), it is natural to model the fairness measures using discrete variables, for which GSVMF (12) paves a generic way;

(iv) For the moderate-sized datasets, we can use exact approaches to solve GSVMF (12) to optimality. For the large-scale optimization, we propose to use the iterative refining strategy (see Section 3), which works extremely well in our numerical study;

(v) If one would like to add the OMR fairness measure as a constraint, the similar results still hold; and

(vi) The OMR fairness measure in GSVMF (12) can be replaced by other fairness measures such as false positive rate (FPR), equal opportunity, and demographic parity, which will be elaborated in Section 4.

We observe that the absolute term in the objective function in GSVMF (12) can be split into two parts by discussing its sign, and therefore GSVMF (12) can be computed by solving two distinct optimization problems. This observation will be particularly important when we design the iterative refining strategy for solving GSVMF (12) in Section 3.

**Proposition 8** *The optimal value of GSVMF (12) is* $v_\rho = \min\{v_{\rho 1}, v_{\rho 2}\}$*, where* $v_{\rho 1}$ *and* $v_{\rho 2}$ *are optimal values of the following two optimization problems*

$$v_{\rho 1} := \min_{\boldsymbol{w}, b, \boldsymbol{u}, \boldsymbol{z}} \frac{1}{N} \sum_{i \in [N]} z_i(u_i - t) + \lambda \|\boldsymbol{w}\|_2^2 - \rho \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} + \rho \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-},$$

$$\text{(GSVMF-1)} \quad \text{s.t.} \sum_{i \in \mathcal{D}_+} D_- z_i - \sum_{i \in \mathcal{D}_-} D_+ z_i \leq 0, \tag{13a}$$

$$(4b) - (4d);$$

$$v_{\rho 2} := \min_{\boldsymbol{w}, b, \boldsymbol{u}, \boldsymbol{z}} \frac{1}{N} \sum_{i \in [N]} z_i(u_i - t) + \lambda \|\boldsymbol{w}\|_2^2 + \rho \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \rho \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-},$$

$$\text{(GSVMF-2)} \quad \text{s.t.} \sum_{i \in \mathcal{D}_+} D_- z_i - \sum_{i \in \mathcal{D}_-} D_+ z_i \geq 0, \tag{13b}$$

$$(4b) - (4d).$$

### 2.4 Connection to Zero-tolerance GSVMF

In this subsection, we observe that if $\rho$ is large enough or more precisely, if there exists a threshold $\bar{\rho}$ such that $\rho \geq \bar{\rho}$, then the fairness measure becomes zero and GSVMF (12)

becomes the following GSVMF0

$$\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \left\{ \frac{1}{N} \sum_{i \in [N]} z_i(u_i - t) + \lambda \|\boldsymbol{w}\|_2^2 : \left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right| = 0, (4b) - (4d) \right\}, \qquad (14)$$

i.e., zero-tolerance GSVMF (12). This result is summarized below.

**Proposition 9** *There exists a positive $\bar{\rho} > 0$ such that for any $\rho \geq \bar{\rho}$, GSVMF reduces to GSVMF0 (14). In particular, $\bar{\rho}$ can be chosen as $\bar{\rho} = \frac{\hat{r}}{\hat{s}}$, where $\hat{r} > 0$ and $\hat{s} > 0$ are defined as below:*

$$\hat{s} := \min_{\boldsymbol{z} \in \{0,1\}^N} \left\{ \left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right| > 0 \right\},$$

$$\hat{r} := \max_{\boldsymbol{z}, \hat{\boldsymbol{z}} \in \{0,1\}^N} \left\{ \min_{\boldsymbol{w},b,\boldsymbol{u}} \left\{ \frac{1}{N} \sum_{i \in [N]} z_i(u_i - t) + \lambda \|\boldsymbol{w}\|_2^2 : (4b) - (4d) \right\} \right.$$

$$\left. - \min_{\boldsymbol{w},b,\boldsymbol{u}} \left\{ \frac{1}{N} \sum_{i \in [N]} \hat{z}_i(u_i - t) + \lambda \|\boldsymbol{w}\|_2^2 : (4b) - (4d) \right\} > 0 \right\}.$$

**Proof** For any $\rho > 0$, let $(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}})$ and $(\boldsymbol{w}^*, b^*, \boldsymbol{u}^*, \boldsymbol{z}^*)$ denote the optimal solutions of GSVMF (12) and GSVMF0 (14), respectively; and let $\widehat{v}_\rho$, $v^*$ denote their corresponding optimal objective values. To prove the result, it is equivalent to show that $\widehat{v}_\rho = v^*$ for any $\rho \geq \bar{\rho}$, i.e., $(\boldsymbol{w}^*, b^*, \boldsymbol{u}^*, \boldsymbol{z}^*)$ is optimal to GSVMF (12).

Since $(\boldsymbol{w}^*, b^*, \boldsymbol{u}^*, \boldsymbol{z}^*)$ is feasible to GSVMF (12), thus we have $v^* \geq \widehat{v}_\rho$. Thus, it remains to show that $v^* \leq \widehat{v}_\rho$ when $\rho \geq \bar{\rho}$, $\bar{\rho} > 0$. We prove it by contradiction. Suppose $v^* > \widehat{v}_\rho$, which implies that $(\boldsymbol{w}^*, b^*, \boldsymbol{u}^*, \boldsymbol{z}^*)$ is not optimal to GSVMF (12), or equivalently, $(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}})$ is not optimal to GSVMF0 (14). Hence, due to optimality condition, we must have

$$\left| \sum_{i \in \mathcal{D}_+} \frac{\widehat{z}_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{\widehat{z}_i}{D_-} \right| > 0. \qquad (15a)$$

Additionally,

$$\frac{1}{N} \sum_{i \in [N]} \widehat{z}_i(\widehat{u}_i - t) + \lambda \|\widehat{\boldsymbol{w}}\|_2^2 = \widehat{v}_\rho - \rho \left| \sum_{i \in \mathcal{D}_+} \frac{\widehat{z}_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{\widehat{z}_i}{D_-} \right| \qquad (15b)$$

$$< v^* - \rho \left| \sum_{i \in \mathcal{D}_+} \frac{\widehat{z}_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{\widehat{z}_i}{D_-} \right| < v^* := \frac{1}{N} \sum_{i \in [N]} z_i^*(u_i^* - t) + \lambda \|\boldsymbol{w}^*\|_2^2,$$

where the first inequality is due to our assumption and the second one is because of (15a).

Then according to (15a) and (15b) as well as the definition of $\hat{s}$ and $\hat{r}$, we must have $\hat{s}$, $\hat{r} > 0$. Hence, we can further conclude that

$$\hat{s} \leq \left| \sum_{i \in \mathcal{D}_+} \frac{\widehat{z}_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{\widehat{z}_i}{D_-} \right|, \qquad (15c)$$

13

$$\frac{1}{N} \sum_{i \in [N]} \widehat{z}_i(\widehat{u}_i - t) + \lambda \|\widehat{\boldsymbol{w}}\|_2^2 \geq \min_{\boldsymbol{w}, b, \boldsymbol{u}} \left\{ \frac{1}{N} \sum_{i \in [N]} \widehat{z}_i(u_i - t) + \lambda \|\boldsymbol{w}\|_2^2 : (4b) - (4d) \right\}$$

$$\geq \frac{1}{N} \sum_{i \in [N]} z_i^*(u_i^* - t) + \lambda \|\boldsymbol{w}^*\|_2^2 - \widehat{r}. \tag{15d}$$

According to (15c) and (15d), we obtain

$$\widehat{v}_\rho = \frac{1}{N} \sum_{i \in [N]} \widehat{z}_i(\widehat{u}_i - t) + \lambda \|\widehat{\boldsymbol{w}}\|_2^2 + \rho \left| \sum_{i \in \mathcal{D}_+} \frac{\widehat{z}_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{\widehat{z}_i}{D_-} \right|$$

$$\geq \frac{1}{N} \sum_{i \in [N]} z_i^*(u_i^* - t) + \lambda \|\boldsymbol{w}^*\|_2^2 - \widehat{r} + \rho \widehat{s}.$$

Thus, for $\rho \geq \bar{\rho} := \frac{\widehat{r}}{\widehat{s}}$, we must have $\widehat{v}_\rho \geq v^*$, a contradiction. ∎

We remark that (i) the proof of Theorem 9 highly relies on the discrete nature of variables $\boldsymbol{z}$, particularly, the reason that two constants $\widehat{r} > 0$ and $\widehat{s} > 0$ are well defined is because binary variables $\boldsymbol{z}$ can only take a finite number of values; and (ii) the result in Theorem 9 shows that by choosing a large but finite penalty $\rho$ instead of being infinite, we are able to convert GSVMF (12) into fairness constrained GSVMF formulation.

## 3. Iterative Refining Strategy (IRS): Theory and Implementations

For the large-scale datasets, exact methods might not be able to solve GSVMF (12) effectively, thus we propose to solve GSVMF (12) using iterative refining strategy (IRS), motivated by alternating minimization method from optimization community. In the IRS, we will optimize binary variables $\boldsymbol{z}$ by fixing the values of the continuous variables $(\boldsymbol{w}, b, \boldsymbol{u})$ in GSVMF (12), termed "unbiased subdata selection," fix the values of the binary variables $\boldsymbol{z}$ and optimize the continuous variables $(\boldsymbol{w}, b, \boldsymbol{u})$; then iterate. In this section, we will study the unbiased subdata selection, design the IRS, and analyze the convergent property of the IRS.

### 3.1 Unbiased Subdata Selection

In this subsection, we will develop a polynomial-time Algorithm 1 with running time complexity $O(N \log N)$ for solving GSVMF (12) by fixing the values of the continuous variables $(\boldsymbol{w}, b, \boldsymbol{u})$. The algorithm takes advantage of the notable result in Theorem 8, that is, we can decompose the problem into two subproblems, GSVMF-1 (13a) and GSVMF-2 (13b). For each subproblem, we observe that if the total number of the correct classification outcomes of the group $g = +1$ is known, i.e., $\sum_{i \in \mathcal{D}_+} z_i = $ constant, then the resulting binary optimization problem can be solved effectively by sorting the objective coefficients. The formal derivation can be found below.

**Proposition 10** *For any fixed $(\boldsymbol{w}, b, \boldsymbol{u})$ in GSVMF (12), optimizing over $\boldsymbol{z}$ can be done in time complexity $O(N \log N)$.*

**Proof** Due to symmetry, it is sufficient to show that the restricted GSVMF-1 (13a) can be solved in $O(N \log N)$ when the values of $(\boldsymbol{w}, b, \boldsymbol{u})$ are fixed. We split the proof into three steps.

**Step 1.** Observe that to solve the restricted GSVMF-1 (13a), we can first enumerate the value of $\sum_{i \in \mathcal{D}_-} z_i$, i.e., we can first let $\sum_{i \in \mathcal{D}_-} z_i = k_-$ for any integer $k_- \in \{0, 1, \ldots, D_-\}$. Then the restricted GSVMF-1 (13a) is equivalent to solve the following two optimization problems

$$\min_{\{z_i\}_{i \in \mathcal{D}_+}} \left\{ \sum_{i \in \mathcal{D}_+} z_i \left[ (u_i - t)/N - \rho/D_+ \right] : \sum_{i \in \mathcal{D}_+} z_i \leq \lfloor k_- D_+/D_- \rfloor \right\}, \tag{16a}$$

$$\min_{\{z_i\}_{i \in \mathcal{D}_-}} \left\{ \sum_{i \in \mathcal{D}_-} z_i \left[ (u_i - t)/N + \rho/D_- \right] : \sum_{i \in \mathcal{D}_-} z_i = k_- \right\}, \tag{16b}$$

where both problems can be solved efficiently via sorting two lists $\{\widehat{u}_i := (u_i - t)/N\}_{i \in \mathcal{D}_+}$ and $\{\widehat{u}_i := (u_i - t)/N\}_{i \in \mathcal{D}_-}$ in the ascending order. Namely, suppose that $\{\widehat{u}_i\}_{i \in \mathcal{D}_+}$ and $\{\widehat{u}_i\}_{i \in \mathcal{D}_-}$ are sorted as $\widehat{u}_{\sigma_+(1)} \leq \ldots \leq \widehat{u}_{\sigma_+(D_+)}$, $\widehat{u}_{\sigma_-(1)} \leq \ldots \leq \widehat{u}_{\sigma_-(D_-)}$, respectively.

Now let us define $\tau_1^* = \arg\max_{i \in \mathcal{D}_+}\{\widehat{u}_i - \rho/D_+ < 0\}$ and $k_+ = \min\{\lfloor k_- D_+/D_- \rfloor, \tau_1^*\}$. Let $z_i^* = 1$ if $i \in \{\sigma_+(\ell)\}_{\ell \in [k_+^*]} \cup \{\sigma_-(\ell)\}_{\ell \in [k_-^*]}$ and 0, otherwise. Then $\{z_i^*\}_{i \in \mathcal{D}_+}, \{z_i^*\}_{i \in \mathcal{D}_-}$ solve the problems (16a) and (16b), respectively.

**Step 2.** We choose the best $\boldsymbol{z}^*$ which achieves the smallest objective value among the possible ones for all $k_- \in \{0, 1, \ldots, D_-\}$.

**Step 3.** Note that the sorting of two lists $\{\widehat{u}_i := (u_i - t)/N\}_{i \in \mathcal{D}_+}$ and $\{\widehat{u}_i := (u_i - t)/N\}_{i \in \mathcal{D}_-}$, and the calculations of $\{\widehat{U}_{\sigma_+}(k_+) := \sum_{i \in [k_+]} \widehat{u}_{\sigma_+(i)}\}_{k_+ \in [D_+]}$ and $\{\widehat{U}_{\sigma_-}(k_-) := \sum_{i \in [k_-]} \widehat{u}_{\sigma_-(i)}\}_{k_- \in [D_-]}$ can be done beforehand with time complexity $O(N \log N)$. Thus, the overall running time complexity is $O(N \log N)$. ∎

The detailed implementation can be found in Algorithm 1. Algorithm 1 is extremely fast when the values of the continuous variables are fixed, which will be illustrated in Section 7.

We remark that (i) the proposed method indeed has the unbiased subdata selection interpretation. The binary variables $\boldsymbol{z}$ indicate fair classification outcomes. Data points with their $z-$ values equal to 1 can be designated as the unbiased selected subdata, using which one can run the classifier to improve classification fairness, as described in the next subsection; and (ii) the subdata selection method can be adapted to solve fairness of black-box classifiers such as convolutional neural network, which will be elaborated in Section 5.

### 3.2 The Proposed Iterative Refining Strategy (IRS)

The proposed IRS is inspired by the alternating minimization method (Chen and Xie, 2020), and the latter is a widely-used approach for solving many machine learning problems such as matrix completion problem (Lai and Varghese, 2017; Jain et al., 2013), and compressive sensing problem (Liao et al., 2014; Abolghasemi et al., 2012). In this subsection, we develop IRS and study its convergence property and approximation bound. To begin with, we let $H(\boldsymbol{w}, b, \boldsymbol{u}, \boldsymbol{z})$ denote the objective function of GSVMF (12). In IRS, we first construct an initial classification solution by solving SVM (2). At each iteration, we solve GSVMF (12) to obtain the optimal $\boldsymbol{z}$ when fixing the continuous variables $(\boldsymbol{w}, b, \boldsymbol{u})$ by letting their

---

**Algorithm 1** Subdata Selection Algorithm for Solving GSVMF with Fixed Continuous Variables

---

**Input:** $\{\boldsymbol{x}_i, y_i\}_{i \in [N]}$, $(\boldsymbol{w}, b, \boldsymbol{u})$, $\rho > 0$, $\lambda > 0$

1: Initialize $k_- = 0$, $v_{c1} = 0$, $v_{c2} = 0$, $k_{1+} = 0$, $k_{2+} = 0$, and $\mathcal{D}_+ = \{i \in [N] : g_i = +1\}$, $\mathcal{D}_- = \{i \in [N] : g_i = -1\}$ with $D_+ = |\mathcal{D}_+|$, $D_- = |\mathcal{D}_-|$
2: Sort $\{\widehat{u}_i = (u_i - t)/N\}_{i \in \mathcal{D}_+}$ and $\{\widehat{u}_i = (u_i - t)/N\}_{i \in \mathcal{D}_-}$ in the ascending order such that $\widehat{u}_{\sigma_+(1)} \leq \dots \leq \widehat{u}_{\sigma_+(D_+)}$, $\widehat{u}_{\sigma_-(1)} \leq \dots \leq \widehat{u}_{\sigma_-(D_-)}$, respectively
3: Calculate $\tau_1^* = \arg\max_{i \in [D_+]}\{\widehat{u}_{\sigma_+(i)} - \rho/D_+ < 0\}$ and $\tau_2^* = \arg\max_{i \in [D_-]}\{\widehat{u}_{\sigma_-(i)} + \rho/D_+ < 0\}$
4: Calculate $\left\{\widehat{U}_{\sigma_+}(k_+) := \sum_{i \in [k_+]} \widehat{u}_{\sigma_+(i)}\right\}_{k_+ \in [D_+]}$ and $\left\{\widehat{U}_{\sigma_-}(k_-) := \sum_{i \in [k_-]} \widehat{u}_{\sigma_-(i)}\right\}_{k_- \in [D_-]}$
5: **do**
6:     **Case 1** Let $k_{1+} = \min\{\lfloor k_- D_+/D_- \rfloor, \tau_1^*\}$, $v_{c1} = \widehat{U}_{\sigma_+}(k_{1+}) - \rho k_{1+}/D_+ + \widehat{U}_{\sigma_-}(k_-) + \rho k_-/D_-$
7:     **Case 2** Let $k_{2+} = \max\{\lceil k_- D_+/D_- \rceil, \tau_2^*\}$, $v_{c2} = \widehat{U}_{\sigma_+}(k_{2+}) + \rho k_{2+}/D_+ + \widehat{U}_{\sigma_-}(k_-) - \rho k_-/D_-$
8:     Let $\ell = \arg\min_{j=1,2}\{v_{cj}\}$, $k_+(k_-) = k_{\ell+}$, $v_{k_-} = v_{c\ell}$, $k_- = k_- + 1$
9: **while** $k_- \leq D_-$
10: Let $k_-^* = \arg\min_{k_- = 0, \dots, D_-}\{v_{k_-}\}$, $k_+^* = k_+(k_-^*)$, $v^* = v_{k_-^*}$, $z_i^* = 1$ if $i \in \{\sigma_+(\ell)\}_{\ell \in [k_+^*]} \cup \{\sigma_-(\ell)\}_{\ell \in [k_-^*]}$, and 0, otherwise

**Output:** $(\boldsymbol{z}^*, v^*)$

---

values be equal to the ones obtained in the previous iteration, and then solve GSVMF (12) to obtain the optimal $(\boldsymbol{w}, b, \boldsymbol{u})$ when fixing the values of binary variables $\boldsymbol{z}$. We continue this procedure until the improvement is within the tolerance. The detailed implementation is described in Algorithm 2.

---

**Algorithm 2** IRS for Solving GSVMF (12)

---

**Input:** $\{\boldsymbol{x}_i, y_i\}_{i \in [N]}$, $\rho > 0$, $\lambda > 0$, $t > 0$, tolerance $\delta > 0$

1: Let $\kappa = 0$, $(\boldsymbol{w}^0, b^0)$ be an optimal solution of SVM (2)
2: Let $u_i^0 = \max\{1 - y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^0 + b^0), 0\}$ and $z_i^0 = \mathbb{I}(u_i^0 \leq t)$ for each $i \in [N]$
3: **do**
4:     Obtain $\boldsymbol{z}^{\kappa+1} \in \arg\min_{\boldsymbol{z} \in \{0,1\}^N} H(\boldsymbol{w}^\kappa, b^\kappa, \boldsymbol{u}^\kappa, \boldsymbol{z})$ using Algorithm 1
5:     Obtain $\left(\boldsymbol{w}^{\kappa+1}, b^{\kappa+1}, \boldsymbol{u}^{\kappa+1}\right) \in \arg\min_{\boldsymbol{w}, b, \boldsymbol{u}}\{H(\boldsymbol{w}, b, \boldsymbol{u}, \boldsymbol{z}^{\kappa+1}) : (4b) - (4c)\}$
6:     $\kappa = \kappa + 1$
7: **while** $H(\boldsymbol{w}^{\kappa-1}, b^{\kappa-1}, \boldsymbol{u}^{\kappa-1}, \boldsymbol{z}^{\kappa-1}) - H(\boldsymbol{w}^\kappa, b^\kappa, \boldsymbol{u}^\kappa, \boldsymbol{z}^\kappa) > \delta$

**Output:** $(\boldsymbol{w}^\kappa, b^\kappa, \boldsymbol{u}^\kappa, \boldsymbol{z}^\kappa)$

---

We remark that (i) one can run Algorithm 2 multiple times if they would like to improve the performance by choosing the initial solution randomly; (ii) Step 4 of IRS Algorithm 2 (i.e., the optimization over $\boldsymbol{z}$) can be effectively solved by employing Algorithm 1; and (iii) the Step 5 of IRS Algorithm 2 (i.e., the optimization over $(\boldsymbol{w}, b, \boldsymbol{u})$) can be effectively solved using first-order methods or existing packages.

Notice that to select the best tuning parameter $\rho$ from a predetermined list using cross-validation, one might need to run IRS Algorithm 2 multiple times. To accelerate this procedure, we can use warm-starts, i.e., the best solution found when solving the previous GSVMF with a different $\rho$ can be set as an initial solution of the current GSVMF. In practice, this warm-start procedure can significantly improve the convergence of the IRS.

Next, we observe that the sequence of the objective values $\{H(\boldsymbol{w}^{\kappa}, b^{\kappa}, \boldsymbol{u}^{\kappa}, \boldsymbol{z}^{\kappa})\}_{\kappa}$ of IRS Algorithm 2 is monotonically non-increasing, is bounded from below, and is thus convergent.

**Proposition 11** *The sequence of the objective values $\{H(\boldsymbol{w}^{\kappa}, b^{\kappa}, \boldsymbol{u}^{\kappa}, \boldsymbol{z}^{\kappa})\}_{\kappa}$ of GSVMF (12) from IRS Algorithm 2 is monotonically non-increasing and bounded from below, and hence converges.*

**Proof** In Algorithm 2, at $(\kappa + 1)$th iteration, given the previous classification decision $\boldsymbol{z}^{\kappa}$, the fact that $(\boldsymbol{w}^{\kappa+1}, b^{\kappa+1}, \boldsymbol{u}^{\kappa+1}) \in \arg\min_{\boldsymbol{w}, b, \boldsymbol{u}} H(\boldsymbol{w}, b, \boldsymbol{u}, \boldsymbol{z}^{\kappa})$ implies that

$$H(\boldsymbol{w}^{\kappa+1}, b^{\kappa+1}, \boldsymbol{u}^{\kappa+1}, \boldsymbol{z}^{\kappa}) \leq H(\boldsymbol{w}^{\kappa}, b^{\kappa}, \boldsymbol{u}^{\kappa}, \boldsymbol{z}^{\kappa}), \tag{17}$$

and $\boldsymbol{z}^{\kappa+1} \in \arg\min_{\boldsymbol{z}} H(\boldsymbol{w}^{\kappa+1}, b^{\kappa+1}, \boldsymbol{u}^{\kappa+1}, \boldsymbol{z})$ implies that

$$H(\boldsymbol{w}^{\kappa+1}, b^{\kappa+1}, \boldsymbol{u}^{\kappa+1}, \boldsymbol{z}^{\kappa+1}) \leq H(\boldsymbol{w}^{\kappa+1}, b^{\kappa+1}, \boldsymbol{u}^{\kappa+1}, \boldsymbol{z}^{\kappa}). \tag{18}$$

Summing up (17) and (18) yields $H(\boldsymbol{w}^{\kappa}, b^{\kappa}, \boldsymbol{u}^{\kappa}, \boldsymbol{z}^{\kappa}) \geq H(\boldsymbol{w}^{\kappa+1}, b^{\kappa+1}, \boldsymbol{u}^{\kappa+1}, \boldsymbol{z}^{\kappa+1})$, which implies that the sequence of output objective values $\{H(\boldsymbol{w}^{\kappa}, b^{\kappa}, \boldsymbol{u}^{\kappa}, \boldsymbol{z}^{\kappa})\}_{\kappa}$ of IRS is monotone non-increasing. On the other hand, according to the definition, we have $H(\boldsymbol{w}^{\kappa}, b^{\kappa}, \boldsymbol{u}^{\kappa}, \boldsymbol{z}^{\kappa}) \geq -t$. Hence, the monotone convergence theorem implies that the sequence $\{H(\boldsymbol{w}^{\kappa}, b^{\kappa}, \boldsymbol{u}^{\kappa}, \boldsymbol{z}^{\kappa})\}_{\kappa}$ is indeed convergent. ∎

Additionally, since $\boldsymbol{z}^{\kappa}$ is binary for all $\kappa$ and there is only a limited number of binary solutions in the set $\{0, 1\}^{N}$, IRS Algorithm 2 will terminate in a finite number of iterations.

We conclude this section by proving the approximation bound of the output IRS Algorithm 2 solution.

**Proposition 12** *Suppose that $(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}})$ denotes a accumulative point of the solution sequence output by IRS Algorithm 2 and $(\boldsymbol{w}^{*}, b^{*}, \boldsymbol{u}^{*}, \boldsymbol{z}^{*})$ denotes an optimal solution of GSVMF (12) with the optimal value $v_{\rho}$. Then we have*

$$v_{\rho} \leq H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}}) \leq v_{\rho} + \frac{M_{u}}{N} |\mathrm{Supp}(\boldsymbol{z}^{*}) \Delta \mathrm{Supp}(\widehat{\boldsymbol{z}})|,$$

*where $v_{\rho} = H(\boldsymbol{w}^{*}, b^{*}, \boldsymbol{u}^{*}, \boldsymbol{z}^{*})$ and $M_{u}$ is defined in Theorem 3.*

**Proof** Recall that we use $H(\boldsymbol{w}, b, \boldsymbol{u}, \boldsymbol{z})$ to denote the objective value of GSVMF (12). We observe that

$$H(\boldsymbol{w}^{*}, b^{*}, \boldsymbol{u}^{*}, \boldsymbol{z}^{*}) \leq H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}}) \leq H(\boldsymbol{w}^{*}, b^{*}, \boldsymbol{u}^{*}, \widehat{\boldsymbol{z}}) \tag{19a}$$

where the first inequality is due to feasibility $(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}})$ of and the second inequality is due to optimality condition of IRS Algorithm 2. Similarly, we also have

$$H(\boldsymbol{w}^{*}, b^{*}, \boldsymbol{u}^{*}, \boldsymbol{z}^{*}) \leq H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}}) \leq H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \boldsymbol{z}^{*}) \tag{19b}$$

17

By summing up (19a) and (19b), we obtain

$$2H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}}) \leq H(\boldsymbol{w}^*, b^*, \boldsymbol{u}^*, \widehat{\boldsymbol{z}}) + H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \boldsymbol{z}^*)$$
$$= \frac{1}{N} \sum_{i \in [N]} (\widehat{z}_i - z_i^*)(u_i^* - \widehat{u}_i) + H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}}) + H(\boldsymbol{w}^*, b^*, \boldsymbol{u}^*, \boldsymbol{z}^*).$$

Thus, we have

$$H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}}) \leq v_\rho + \frac{1}{N} \sum_{i \in [N]} (\widehat{z}_i - z_i^*)(u_i^* - \widehat{u}_i) \leq v_\rho + \frac{1}{N} \sum_{i \in \text{Supp}(\boldsymbol{z}^*)\Delta\text{Supp}(\widehat{\boldsymbol{z}})} |u_i^* - \widehat{u}_i|, \quad (20)$$

where the second inequality is because of $u_i^* - \widehat{u}_i \leq |u_i^* - \widehat{u}_i|$ for each $i \in [N]$. Following the proof of Theorem 3, without loss of generality, we can restrict $0 \leq u_i^*, \widehat{u}_i \leq M_u$ for any $i \in [N]$. Thus, we further have

$$H(\widehat{\boldsymbol{w}}, \widehat{b}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{z}}) \leq v_\rho + \frac{M_u}{N} |\text{Supp}(\boldsymbol{z}^*)\Delta\text{Supp}(\widehat{\boldsymbol{z}})|.$$

This completes the proof. ∎

We remark that (i) the solution-quality of IRS Algorithm 2 depends on the difference between the binary variables (i.e., the subdata selection decisions) $\widehat{\boldsymbol{z}}$ and $\boldsymbol{z}^*$. This is not surprising since the subdata selection decisions are indeed crucial to GSVMF (12) and are responsible for balancing the classification accuracy and fairness; and (ii) Since $M_u$ is independent from the number of data points $N$, thus if $|\text{Supp}(\boldsymbol{z}^*)\Delta\text{Supp}(\widehat{\boldsymbol{z}})| = o(N)$, then IRS Algorithm 2 is asymptotically optimal when $N \to \infty$. This further demonstrates the effectiveness of the proposed solution algorithm.

## 4. Variation I: Different Fairness Measures

In this section, we demonstrate the following variations of the proposed GSVMF (12) with the different fairness measures such as false positive rate (FPR), equal opportunity (a.k.a., false negative rate), and demographic parity. All the formulations can be solved by the proposed IRS Algorithm 2 effectively.

### 4.1 False Positive Rate Fairness

False positive rate (FPR) fairness (see, e.g., Zafar et al. 2019) is another misclassification measure of disparate mistreatment. A classifier has no FPR disparate mistreatment if the different protected groups with label $y = -1$ have the same misclassification rates. More formally, FPR fairness measure can be defined as follows.

**Definition 13** *Given a threshold $t > 0$, the FPR fairness measure for the binary classification is*

$$F_{\text{FPR}} = |\mathbb{P}(\tilde{u} \leq t | \tilde{g} = +1, \tilde{y} = -1) - \mathbb{P}(\tilde{u} \leq t | \tilde{g} = -1, \tilde{y} = -1)|,$$

*where $\tilde{u}$ denotes violation margin.*

In Theorem 13, $F_{\mathrm{FPR}}$ denotes the absolute difference of the misclassification rates of both groups with label $\tilde{y} = -1$. Using the binary variables $\boldsymbol{z}$, the fairness measure $F_{\mathrm{FPR}}$ can be computed by

$$F_{\mathrm{FPR}} = |\mathbb{P}(\tilde{u} \leq t | \tilde{g} = +1, \tilde{y} = -1) - \mathbb{P}(\tilde{u} \leq t | \tilde{g} = -1, \tilde{y} = -1)| = \left| \sum_{i \in \mathcal{D}_{+-}} \frac{z_i}{D_{+-}} - \sum_{i \in \mathcal{D}_{--}} \frac{z_i}{D_{--}} \right|,$$

$$(21\mathrm{a})$$

where two sets

$$\mathcal{D}_{+-} = \{i \in [N] : g_i = +1, y_i = -1\}, \mathcal{D}_{--} = \{i \in [N] : g_i = -1, y_i = -1\}, \qquad (21\mathrm{b})$$

represent collections of indices of data points from each protected group with true negative labels $y = -1$, and $D_{+-} = |\mathcal{D}_{+-}|$ is the number of group $g = +1$ data points with label $y = -1$, and $D_{--} = |\mathcal{D}_{--}|$ is the number of group $g = -1$ data points with label $y = -1$.

With this observation, we can replace the OMR fairness measure $\mathcal{F}(\boldsymbol{z})$ in GSVMF (12) by FPR fairness as follows

$$\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \left\{ \underbrace{\frac{1}{N} \sum_{i \in [N]} z_i(u_i - t)}_{\mathcal{A}(\boldsymbol{z},t\boldsymbol{e}-\boldsymbol{u})} + \lambda \underbrace{\|\boldsymbol{w}\|_2^2}_{\mathcal{R}(\boldsymbol{w},b,\boldsymbol{u})} + \rho \underbrace{\left| \sum_{i \in \mathcal{D}_{+-}} \frac{z_i}{D_{+-}} - \sum_{i \in \mathcal{D}_{--}} \frac{z_i}{D_{--}} \right|}_{\mathcal{F}(\boldsymbol{z})} : (4\mathrm{b}) - (4\mathrm{d}) \right\}. \quad (22)$$

### 4.2 Equal Opportunity Fairness

Equal opportunity fairness is a popular fairness measure in which the prediction is independent of the sensitive feature for the data points with positive label (see, e.g., Olfat and Aswani 2017). It is worthy of mentioning that equal opportunity fairness is also known as false negative rate fairness. Formally, equal opportunity fairness measure has the following representation.

**Definition 14** *Given a threshold $t > 0$, the equal opportunity fairness measure for the binary classification is*

$$F_{\mathrm{EO}} = |\mathbb{P}(\tilde{u} \leq t | \tilde{g} = +1, \tilde{y} = +1) - \mathbb{P}(\tilde{u} \leq t | \tilde{g} = -1, \tilde{y} = +1)|,$$

*where $\tilde{u}$ denotes violation margin.*

In Theorem 14, $F_{\mathrm{EO}}$ denotes the absolute difference of the probabilities of predicting positive labels for group $g = +1$ and group $g = -1$ with label $\tilde{y} = +1$. That is, in this definition, the positive label $y = +1$ is viewed to be more important than its negative counterpart $y = -1$. Using the binary variables $\boldsymbol{z}$, the fairness measure $F_{\mathrm{EO}}$ can be computed as

$$F_{\mathrm{EO}} = |\mathbb{P}(\tilde{u} \leq t | \tilde{g} = +1, \tilde{y} = +1) - \mathbb{P}(\tilde{u} \leq t | \tilde{g} = -1, \tilde{y} = +1)| = \left| \sum_{i \in \mathcal{D}_{++}} \frac{z_i}{D_{++}} - \sum_{i \in \mathcal{D}_{-+}} \frac{z_i}{D_{-+}} \right|,$$

$$(23\mathrm{a})$$

where two sets

$$\mathcal{D}_{++} = \{i \in [N] : g_i = +1, y_i = +1\}, \mathcal{D}_{-+} = \{i \in [N] : g_i = -1, y_i = +1\} \quad (23b)$$

represent collections of indices of data points from each protected group with label $y = +1$, and $D_{++} = |\mathcal{D}_{++}|$ is the number of group $g = +1$ data points with label $y = +1$, and $D_{-+} = |\mathcal{D}_{-+}|$ is the number of group $g = -1$ data points with label $y = +1$.

With this observation, we can replace the OMR fairness measure $\mathcal{F}(\boldsymbol{z})$ in GSVMF (12) by equal opportunity fairness as follows

$$\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \left\{ \underbrace{\frac{1}{N}\sum_{i\in[N]} z_i(u_i - t)}_{\mathcal{A}(\boldsymbol{z},t\boldsymbol{e}-\boldsymbol{u})} + \lambda \underbrace{\|\boldsymbol{w}\|_2^2}_{\mathcal{R}(\boldsymbol{w},b,\boldsymbol{u})} + \rho \underbrace{\left| \sum_{i\in\mathcal{D}_{++}} \frac{z_i}{D_{++}} - \sum_{i\in\mathcal{D}_{-+}} \frac{z_i}{D_{-+}} \right|}_{\mathcal{F}(\boldsymbol{z})} : (4b) - (4d) \right\}. \quad (24)$$

### 4.3 Demographic Parity Fairness

Demographic parity fairness is one of the common fairness measures in which the prediction is independent of the sensitive feature (see, e.g., Olfat and Aswani 2017). The formal definition of demographic parity fairness can be found as below.

**Definition 15** *Given a threshold $t > 0$, the demographic parity fairness measure for the binary classification is*

$$F_{\mathrm{DP}} = |\mathbb{P}(\tilde{u} \le t, \tilde{y} = +1|\tilde{g} = +1) + \mathbb{P}(\tilde{u} > t, \tilde{y} = -1|\tilde{g} = +1)$$
$$-\mathbb{P}(\tilde{u} \le t, \tilde{y} = +1|\tilde{g} = -1) - \mathbb{P}(\tilde{u} > t, \tilde{y} = -1|\tilde{g} = -1)|,$$

*where $\tilde{u}$ denotes violation margin.*

In Theorem 15, $F_{\mathrm{DP}}$ denotes the absolute difference of the probabilities of predicting positive labels for group $g = +1$ and group $g = -1$. For the binary classification, we observe that a data point in class $y = +1$ will have positive predicted label if the prediction is correct, and a data point in class $y = -1$ will have positive predicted label if the prediction is incorrect. Based on this observation, we can use the binary variables $\boldsymbol{z}$ and their complements $\boldsymbol{e} - \boldsymbol{z}$ to indicate the positive predicted labels from class $y = +1$ and class $y = -1$ in GSVM (4), respectively. Using the binary variables $\boldsymbol{z}$, the fairness measure $F_{\mathrm{DP}}$ can be computed by

$$F_{\mathrm{DP}} = |\mathbb{P}(\tilde{u} \le t, \tilde{y} = +1|\tilde{g} = +1) + \mathbb{P}(\tilde{u} > t, \tilde{y} = -1|\tilde{g} = +1)$$
$$-\mathbb{P}(\tilde{u} \le t, \tilde{y} = +1|\tilde{g} = -1) - \mathbb{P}(\tilde{u} > t, \tilde{y} = -1|\tilde{g} = -1)|$$
$$= \left| \frac{\sum_{i\in\mathcal{D}_{++}} z_i + \sum_{i\in\mathcal{D}_{+-}}(1 - z_i)}{D_+} - \frac{\sum_{i\in\mathcal{D}_{-+}} z_i + \sum_{i\in\mathcal{D}_{--}}(1 - z_i)}{D_-} \right|, \quad (25)$$

where the sets $\mathcal{D}_{++}, \mathcal{D}_{+-}, \mathcal{D}_{-+}, \mathcal{D}_{--}$ are defined in (21b) and (23b), and $D_+ = |\mathcal{D}_+|$ is the number of group $g = +1$ data points, and $D_- = |\mathcal{D}_-|$ is the number of group $g = -1$ data points.

With this observation, we can replace the OMR fairness measure $\mathcal{F}(\boldsymbol{z})$ in GSVMF (12) by demographic parity fairness as follows

$$
\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \left\{ \underbrace{\frac{1}{N}\sum_{i\in[N]} z_i(u_i-t)}_{\mathcal{A}(\boldsymbol{z},t\boldsymbol{e}-\boldsymbol{u})} + \lambda \underbrace{\|\boldsymbol{w}\|_2^2}_{\mathcal{R}(\boldsymbol{w},b,\boldsymbol{u})} \right.
$$

$$
\left. +\rho \underbrace{\left| \frac{\sum_{i\in\mathcal{D}_{++}} z_i + \sum_{i\in\mathcal{D}_{+-}}(1-z_i)}{D_+} - \frac{\sum_{i\in\mathcal{D}_{-+}} z_i + \sum_{i\in\mathcal{D}_{--}}(1-z_i)}{D_-} \right|}_{\mathcal{F}(\boldsymbol{z})} : (4b)-(4d) \right\}. \tag{26}
$$

## 5. Variation II: Different Classifiers and Fairness Measures

In this section, we demonstrate the following variations of the proposed framework (1) with the generalized OMR fairness measure: (i) We develop the fair multiclass classification formulation; (ii) We propose the generalized logistic regression formulation with fairness; (iii) We study the generalized kernel SVM formulation with fairness; and (iv) We incorporate the OMR fairness into deep learning models.

### 5.1 Fair Multiclass Classification

If there are multiple classes, GSVMF (12) can be modified as the following generalized multiclass SVM with the OMR fairness (GMSVMF)

$$
v_\rho^M := \min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \underbrace{\frac{1}{N}\sum_{i\in[N]}\sum_{j\in[K]:j\neq y_i}(1-z_{ij})(u_{ij}-t)}_{\mathcal{A}(\boldsymbol{z},t\boldsymbol{e}-\boldsymbol{u})} + \lambda\underbrace{\sum_{j\in[K]}\|\boldsymbol{w}_j\|_2^2}_{\mathcal{R}(\boldsymbol{w},b,\boldsymbol{u})} + \rho\underbrace{\left|\sum_{i\in\mathcal{D}_+}\frac{z_{iy_i}}{D_+} - \sum_{i\in\mathcal{D}_-}\frac{z_{iy_i}}{D_-}\right|}_{\mathcal{F}(\boldsymbol{z})},
$$

$$\tag{27a}$$

(GMSVMF) s.t. $\boldsymbol{w}_{y_i}^\top \boldsymbol{x}_i + b_{y_i} - \boldsymbol{w}_j^\top \boldsymbol{x}_i - b_j \geq 1 - u_{ij}, \forall i \in [N], j \in [K] : j \neq y_i,$ $\tag{27b}$

$$
\sum_{j\in[K]} z_{ij} = 1, \forall i \in [N], \tag{27c}
$$

$$
u_{ij} \geq 0, z_{ij} \in \{0,1\}, \forall i \in [N], j \in [K], \tag{27d}
$$

where $D_+ = \sum_{i\in[N]}\mathbb{1}(g_i = +1)$ and $D_- = \sum_{i\in[N]}\mathbb{1}(g_i = -1)$ are the numbers of the protected groups, and $K \geq 2$ represents the number of classes. The proposed GMSVMF (27) is a special case of the framework (1). For each $i \in [N], j \in [K]$, let binary variable $z_{ij} = 1$ if we predict $i$th data point to be label $j$, and 0, otherwise. Constraints (27c) enforce the classifier to assign exact one label to each instance. When $K = 2$, GMSVMF (27) becomes a binary classification problem with the OMR fairness, where $j = 1$ denotes the positive label and $j = 2$ denotes the negative label. Note that the notion of OMR fairness

measure (i.e., Theorem 6) is simply extended to the multi-class classification model, which defines the absolute difference of the misclassification rates of both groups.

Note that GMSVMF (27) has non-convex terms $\{z_{ij}u_{ij}\}_{i\in[N],j\in[K]:j\neq y_i}$, which can be linearized using the similar McCormick inequalities as (5). That is, we introduce new variables $s_{ij} = z_{ij}u_{ij}$ for each $i \in [N], j \in [K] : j \neq y_i$ and then linearize it as

$$s_{ij} \leq M_u z_{ij}, \quad s_{ij} \leq u_{ij}, \quad s_{ij} \geq 0, \quad s_{ij} \geq u_{ij} - M_u(1 - z_{ij}), \forall i \in [N], j \in [K] : j \neq y_i. \quad (28)$$

Then we can obtain an equivalent MICP formulation.

**Proposition 16** *GMSVMF (27) can be recast as the following MICP*

$$v_\rho^M = \min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z},\boldsymbol{s}} \left\{ \frac{1}{N} \sum_{i\in[N]} \sum_{j\in[K]:j\neq y_i} (u_{ij} - t - s_{ij} + z_{ij}t) + \lambda \sum_{j\in[K]} \|\boldsymbol{w}_j\|_2^2 \right.$$
$$\left. +\rho \left| \sum_{i\in\mathcal{D}_+} \frac{z_{iy_i}}{D_+} - \sum_{i\in\mathcal{D}_-} \frac{z_{iy_i}}{D_-} \right| : (27\text{b}) - (27\text{d}), (28) \right\}.$$

Similar to GSVMF (12), we can decompose the fair multiclass classification problem into two subproblems, GMSVMF-1 and GMSVMF-2 by discussing the sign of the absolute function.

**Proposition 17** *The optimal value of GMSVMF (27) is $v_\rho^M = \min\{v_{\rho 1}^M, v_{\rho 2}^M\}$, where $v_{\rho 1}^M$ and $v_{\rho 2}^M$ are optimal values of the following two optimization problems*

$$v_{\rho 1}^M := \min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \frac{1}{N} \sum_{i\in[N]} \sum_{j\in[K]:j\neq y_i} (1 - z_{ij})(u_{ij} - t) + \lambda \sum_{j\in[K]} \|\boldsymbol{w}_j\|_2^2 - \rho \sum_{i\in\mathcal{D}_+} \frac{z_{iy_i}}{D_+} + \rho \sum_{i\in\mathcal{D}_-} \frac{z_{iy_i}}{D_-},$$

(GMSVMF-1) *s.t.* $\sum_{i\in\mathcal{D}_+} D_- z_{iy_i} - \sum_{i\in\mathcal{D}_-} D_+ z_{iy_i} \leq 0,$

$$(27\text{b}) - (27\text{d});$$

$$(29\text{a})$$

$$v_{\rho 2}^M := \min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \frac{1}{N} \sum_{i\in[N]} \sum_{j\in[K]:j\neq y_i} (1 - z_{ij})(u_{ij} - t) + \lambda \sum_{j\in[K]} \|\boldsymbol{w}_j\|_2^2 + \rho \sum_{i\in\mathcal{D}_+} \frac{z_{iy_i}}{D_+} - \rho \sum_{i\in\mathcal{D}_-} \frac{z_{iy_i}}{D_-},$$

(GMSVMF-2) *s.t.* $\sum_{i\in\mathcal{D}_+} D_- z_{iy_i} - \sum_{i\in\mathcal{D}_-} D_+ z_{iy_i} \geq 0,$

$$(27\text{b}) - (27\text{d}).$$

$$(29\text{b})$$

Next, we show that when fixing the values of the continuous variables $(\boldsymbol{w}, b, \boldsymbol{u})$, the GMSVMF (27) can be solved efficiently.

**Proposition 18** *For any fixed $(\boldsymbol{w}, b, \boldsymbol{u})$ in GMSVMF (27), optimizing over $\boldsymbol{z}$ can be done in time complexity $O(\max\{NK, N \log N\})$.*

**Proof** Due to symmetry, it is sufficient to show that the restricted GMSVMF-1 (29a) can be solved in $O(N \log N)$ when the values of $(\boldsymbol{w}, b, \boldsymbol{u})$ are fixed. We split the proof into three steps.

**Step 1.** Observe that to solve the restricted GMSVMF-1 (29a), we can first enumerate the value of $\sum_{i \in \mathcal{D}_-} z_{iy_i}$, i.e., we can let $\sum_{i \in \mathcal{D}_-} z_{iy_i} = k_-$ for any integer $k_- \in \{0, 1, \ldots, D_-\}$.

For each $i \in [N]$, by discussing whether $z_{iy_i} = 1$ or not, the restricted GMSVMF-1 (29a) with $\sum_{i \in \mathcal{D}_-} z_{iy_i} = k_-$ is equivalent to solve the following two optimization problems

$$\min_{\{z_i\}_{i \in \mathcal{D}_+}} \left\{ \sum_{i \in \mathcal{D}_+} \left[ z_{iy_i} \left[ \sum_{j \in [K]: j \neq y_i} (u_{ij} - t)/N - \rho/D_+ \right] + (1 - z_{iy_i}) \sum_{j \in [K]: j \neq y_i, j \neq j_i^*} (u_{ij} - t)/N \right] : \right.$$
$$\left. \sum_{i \in \mathcal{D}_+} z_{iy_i} \leq \lfloor k_- D_+/D_- \rfloor \right\},$$
(30a)

$$\min_{\{z_i\}_{i \in \mathcal{D}_-}} \left\{ \sum_{i \in \mathcal{D}_-} \left[ z_{iy_i} \left[ \sum_{j \in [K]: j \neq y_i} (u_{ij} - t)/N + \rho/D_- \right] + (1 - z_{iy_i}) \sum_{j \in [K]: j \neq y_i, j \neq j_i^*} (u_{ij} - t)/N \right] : \right.$$
$$\left. \sum_{i \in \mathcal{D}_-} z_{iy_i} = k_- \right\},$$
(30b)

where $j_i^* \in \arg\max_{j \in [K]: j \neq y_i} \{u_{ij}\}$ denotes the incorrect label that will be predicted for each $i$ if the data point is misclassified. Both problems can be solved efficiently via sorting two lists $\{\widehat{c}_i = \bar{c}_{i1} - \bar{c}_{i0}\}_{i \in \mathcal{D}_+}$ and $\{\widehat{c}_i = \bar{c}_{i1} - \bar{c}_{i0}\}_{i \in \mathcal{D}_-}$ in the ascending order, where $\bar{c}_{i1} = \sum_{j \in [K]: j \neq y_i} (u_{ij} - t)/N$, and $\bar{c}_{i0} = \sum_{j \in [K]: j \neq y_i, j \neq j_i^*} (u_{ij} - t)/N$ are the costs for correct classification and misclassification, respectively.

In particular, suppose that $\{\widehat{c}_i\}_{i \in \mathcal{D}_+}$ and $\{\widehat{c}_i\}_{i \in \mathcal{D}_-}$ are sorted as $\widehat{c}_{\sigma_+(1)} \leq \ldots \leq \widehat{c}_{\sigma_+(D_+)}$, $\widehat{c}_{\sigma_-(1)} \leq \ldots \leq \widehat{c}_{\sigma_-(D_-)}$, respectively. Now let $\tau_1^* = \arg\max_{i \in [D_+]} \{\widehat{c}_{\sigma_+(i)} - \rho/D_+ < 0\}$, and $k_+ = \min\{\lfloor k_- D_+/D_- \rfloor, \tau_1^*\}$. Then let $z_{ij}^* = 1$ if $i \in \{\sigma_+(\ell)\}_{\ell \in [k_+^*]} \cup \{\sigma_-(\ell)\}_{\ell \in [k_-^*]}, j = y_i$, or $i \notin \{\sigma_+(\ell)\}_{\ell \in [k_+^*]} \cup \{\sigma_-(\ell)\}_{\ell \in [k_-^*]}, j = j_i^*$, and 0, otherwise. Then $\{z_{ij}^*\}_{i \in \mathcal{D}_+}, \{z_{ij}^*\}_{i \in \mathcal{D}_-}$ solve the problems (30a) and (30b), respectively.

**Step 2.** Then we choose the best $\boldsymbol{z}^*$ which achieves the smallest objective value among the possible ones for all $k_- \in \{0, 1, \ldots, D_-\}$.

**Step 3.** Note that the sortings of two lists $\{\widehat{c}_i = \bar{c}_{i1} - \bar{c}_{i0}\}_{i \in \mathcal{D}_+}$ and $\{\widehat{c}_i = \bar{c}_{i1} - \bar{c}_{i0}\}_{i \in \mathcal{D}_-}$ can be done beforehand with time complexity $O(N \log N)$, while computing $\{j_i^*\}_{i \in [N]}$ takes $O(NK)$ time. Thus, the overall running time complexity is $O(\max\{NK, N \log N\})$. ∎

The detailed implementation can be found in Algorithm 3. Finally, we recommend using IRS Algorithm 2 to solve GMSVMF (27), wherein Step 4, using Algorithm 3 instead of Algorithm 1.

---

**Algorithm 3** Subdata Selection Algorithm for Solving GMSVMF (27) with Fixed Continuous Variables

---

**Input:** $\{\boldsymbol{x}_i, y_i\}_{i \in [N]}$, $(\boldsymbol{w}, b, \boldsymbol{u})$, $\rho > 0$, $\lambda > 0$

1: Initialize $k_- = 0$, $k_{1+} = 0$, $k_{2+} = 0$, and $\mathcal{D}_+ = \{i \in [N] : g_i = +1\}, \mathcal{D}_- = \{i \in [N] : g_i = -1\}$ with $D_+ = |\mathcal{D}_+|$, $D_- = |\mathcal{D}_-|$

2: Let $j_i^* = \arg\max_{j \in [K]: j \neq y_i}\{u_{ij}\}$, $\bar{c}_{i1} = \sum_{j \in [K]: j \neq y_i}(u_{ij} - t)/N$, $\bar{c}_{i0} = \sum_{j \in [K]: j \neq y_i, j \neq j_i^*}(u_{ij} - t)/N$ for each $i \in [N]$. Sort $\{\hat{c}_i = \bar{c}_{i1} - \bar{c}_{i0}\}_{i \in \mathcal{D}_+}$ and $\{\hat{c}_i = \bar{c}_{i1} - \bar{c}_{i0}\}_{i \in \mathcal{D}_-}$ in the ascending order such that $\hat{c}_{\sigma_+(1)} \leq ... \leq \hat{c}_{\sigma_+(D_+)}$, $\hat{c}_{\sigma_-(1)} \leq ... \leq \hat{c}_{\sigma_-(D_-)}$, respectively

3: Let $\tau_1^* = \arg\max_{i \in [D_+]}\{\hat{c}_{\sigma_+(i)} - \rho/D_+ < 0\}$, $\tau_2^* = \arg\max_{i \in [D_-]}\{\hat{c}_{\sigma_-(i)} + \rho/D_+ < 0\}$, $v_{c1} = \sum_{i \in [N]} \hat{c}_{i0}$ and $v_{c2} = \sum_{i \in [N]} \hat{c}_{i0}$

4: Calculate $\left\{\hat{C}_{\sigma_+}(k_+) := \sum_{i \in [k_+]} \hat{c}_{\sigma_+(i)}\right\}_{k_+ \in [D_+]}$ and $\left\{\hat{C}_{\sigma_-}(k_-) := \sum_{i \in [k_-]} \hat{c}_{\sigma_-(i)}\right\}_{k_- \in [D_-]}$

5: **do**

6:     **Case 1** Let $k_{1+} = \min\{\lfloor k_- D_+/D_- \rfloor, \tau_1^*\}$, $v_{c1} = \hat{C}_{\sigma_+}(k_{1+}) - \rho k_{1+}/D_+ + \hat{C}_{\sigma_-}(k_-) + \rho k_-/D_-$

7:     **Case 2** Let $k_{2+} = \max\{\lceil k_- D_+/D_- \rceil, \tau_2^*\}$, $v_{c2} = \hat{C}_{\sigma_+}(k_{2+}) + \rho k_{2+}/D_+ + \hat{C}_{\sigma_-}(k_-) - \rho k_-/D_-$

8:     Let $\ell = \arg\min_{j=1,2}\{v_{cj}\}$, $k_+(k_-) = k_{\ell+}$, $v_{k_-} = v_{c\ell}$, $k_- = k_- + 1$

9: **while** $k_- \leq D_-$

10: Let $k_-^* = \arg\min_{k_-=0,...,D_-}\{v_{k_-}\}$, $k_+^* = k_+(k_-^*)$, $v^* = v_{k_-^*}$, $z_{ij}^* = 1$ if $i \in \{\sigma_+(\ell)\}_{\ell \in [k_+^*]} \cup \{\sigma_-(\ell)\}_{\ell \in [k_-^*]}, j = y_i$, or $i \notin \{\sigma_+(\ell)\}_{\ell \in [k_+^*]} \cup \{\sigma_-(\ell)\}_{\ell \in [k_-^*]}, j = j_i^*$, and 0, otherwise

**Output:** $(\boldsymbol{z}^*, v^*)$

---

## 5.2 Generalized Logistic Regression with Fairness

Logistic regression is another popular binary classification method, which has been used in many areas such as medical data classification (Dreiseitl and Ohno-Machado, 2002), susceptibility mapping (Ayalew and Yamagishi, 2005), and fraud detection (Shen et al., 2007). Given a dataset $\{(\boldsymbol{x}_i, y_i)\}_{i \in [N]} \subseteq \mathbb{R}^n \times \{0, 1\}$ with $N$ data points, following the similar derivation of GSVMF (12), we propose the generalized logistic regression with fairness (GLRF) as follows

$$\min_{\boldsymbol{w}, b, \boldsymbol{z} \in \{0,1\}^N} \underbrace{\frac{1}{N} \sum_{i \in [N]} z_i \left(t - u_i\right)}_{\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})} + \lambda \underbrace{\|\boldsymbol{w}\|_2^2}_{\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u})} + \rho \underbrace{\left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right|}_{\mathcal{F}(\boldsymbol{z})}, \tag{31a}$$

$$\text{s.t.} \quad u_i = y_i \log \left(h_{(\boldsymbol{w}, b)}(\boldsymbol{x_i})\right) + (1 - y_i) \log \left(1 - h_{(\boldsymbol{w}, b)}(\boldsymbol{x_i})\right), \forall i \in [N], \tag{31b}$$

where $h_{(\boldsymbol{w}, b)}(\boldsymbol{x}_i) = [1 + \exp(-\boldsymbol{x}_i^\top \boldsymbol{w} - b)]^{-1}$ denotes the sigmoid function.

We remark that (i) GLRF (31) can be extended to incorporate other fairness measures in Section 4; and (ii) GLRF (31) and its variants can be solved by the IRS Algorithm 2 as well.

## 5.3 Generalized Kernel SVM with Fairness

When the data are not suitable for linear models, kernel SVM (KSVM) can be a better alternative and can map the nonlinear models into a higher dimensional space. It is worth mentioning that the choice of kernel functions and parameters might affect the capability of the classifiers (Han et al., 2012). To incorporate the fairness measure in KSVM, we propose the following GKSVMF formulation:

$$\min_{\boldsymbol{z}} \left\{ K(\boldsymbol{z}, \boldsymbol{u}) := \underbrace{\frac{1}{N} \sum_{i \in [N]} z_i (u_i - t)}_{\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})} + \rho \underbrace{\left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right|}_{\mathcal{F}(\boldsymbol{z})} : z_i \in \{0, 1\}, \forall i \in [N] \right\}, \tag{32}$$

where $\boldsymbol{u}$ is the vector of violation margins. Note that the proposed GKSVMF (32) is also a special case of the framework (1) with $\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u}) = 0$.

The IRS Algorithm 2 can be adapted to solve GKSVMF (32). We first obtain the initial solution from KSVM. We solve GKSVMF (32) to select the unbiased data points with the given prediction margins $\boldsymbol{u}$, and then train KSVM with the selected subdata. We continue this procedure until no improvement or other stopping criteria being invoked. The detailed implementation is described in Algorithm 4.

We remark that (i) Step 5 in Algorithm 4 can be replaced by other black-box classifiers; (ii) GKSVMF (32) can be adapted to incorporate other fairness measures in Section 4; and (iii) GKSVMF (32) and its variants can be solved by Algorithm 4 as well.

---

**Algorithm 4** IRS for Solving GKSVMF (32)

---

**Input:** $\{\boldsymbol{x}_i, y_i\}_{i \in [N]}$, $\rho > 0$, $t > 0$

1: Let $\kappa = 0$, $\boldsymbol{u}^0$ be an optimal solution of $KSVM(\{\boldsymbol{x}_i, y_i\}_{i \in [N]})$
2: Let $z_i^0 = \mathbb{I}(u_i^0 \leq t)$ for each $i \in [N]$
3: **do**
4:     Obtain $\boldsymbol{z}^{\kappa+1} \in \arg\min_{\boldsymbol{z} \in \{0,1\}^N} K(\boldsymbol{z}, \boldsymbol{u}^\kappa)$
5:     Obtain $\boldsymbol{u}^{\kappa+1} \in \arg\min_{\boldsymbol{u}} KSVM(\{\boldsymbol{x}_i, y_i\}_{i \in [N]: z_i^{\kappa+1}=1})$
6:     $\kappa = \kappa + 1$
7: **while** $K(\boldsymbol{z}^{\kappa-1}, \boldsymbol{u}^{\kappa-1}) - K(\boldsymbol{z}^\kappa, \boldsymbol{u}^\kappa) > 0$

**Output:** $(\boldsymbol{z}^\kappa, \boldsymbol{u}^\kappa)$

---

### 5.4 Fair Deep Learning

Convolutional neural network (CNN) is a popular method for image classification in deep learning (O'Shea and Nash, 2015). The Vanilla CNN (VCNN) might be biased against some protected groups. For example, in medical image diagnosis, VCNN model was reported to show discrimination against the gender (Du et al., 2020). As far as we are concerned, there is no systematic way to deal with the fairness of VCNN.

Similar to GKSVMF (32), using the binary variables $\boldsymbol{z}$ to represent the subdata selection decisions, we propose the following formulation for Fair CNN (FCNN):

$$\min_{\boldsymbol{z} \in \{0,1\}^N} \left\{ C(\boldsymbol{z}, \boldsymbol{u}) := \underbrace{\frac{1}{N} \sum_{i \in [N]} z_i (t - u_i)}_{\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})} + \rho \underbrace{\left| \sum_{i \in \mathcal{D}_+} \frac{z_i}{D_+} - \sum_{i \in \mathcal{D}_-} \frac{z_i}{D_-} \right|}_{\mathcal{F}(\boldsymbol{z})} \right\}, \tag{33}$$

Above, vector $\boldsymbol{u}$ is the output of VCNN, where $u_i$ denotes the probability of correctly predicting the label for data point $i \in [N]$. Different from GKSVMF (32), we use $t \in [0, 1]$ to denote the prediction probability threshold, i.e., for each data point $i \in [N]$, $z_i = 1$ if the classification probability $u_i \geq t$, and 0, otherwise. That is, for each data point $i \in [N]$, $z_i = 1$ indicates the correct prediction when the classification probability $u_i \geq t$ and $z_i = 0$ indicates the incorrect prediction. The proposed FCNN (33) is also a special case of the framework (1) with $\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u}) = 0$.

The IRS Algorithm 2 can be adapted to solve FCNN (33). We first construct the initial classification solution by solving VCNN. We solve FCNN (33) to select the unbiased subdata, and then train VCNN with the subdata. We continue this procedure until the improvement is within the tolerance. The detailed implementation is described in Algorithm 5.

We remark that (i) Step 5 in Algorithm 5 can be replaced by other deep learning models or black-box classifiers; (ii) FCNN (33) can be extended to incorporate other fairness measures in Section 4; and (iii) FCNN (33) and its variants can be solved effectively by Algorithm 5.

---

**Algorithm 5** IRS for Solving FCNN (33)

---

**Input:** $\{\boldsymbol{x}_i, y_i\}_{i \in [N]}$, $\rho > 0$, $t > 0$

1: Let $\kappa = 0$, $\boldsymbol{u}^0$ be an optimal solution of $VCNN(\{\boldsymbol{x}_i, y_i\}_{i \in [N]})$
2: Let $z_i^0 = \mathbb{I}(t \leq u_i^0)$ for each $i \in [N]$
3: **do**
4:    Obtain $\boldsymbol{z}^{\kappa+1} \in \arg\min_{\boldsymbol{z} \in \{0,1\}^N} C(\boldsymbol{z}, \boldsymbol{u}^\kappa)$
5:    Obtain $\boldsymbol{u}^{\kappa+1} \in \arg\min_{\boldsymbol{u}} VCNN(\{\boldsymbol{x}_i, y_i\}_{i \in [N]: z_i^{\kappa+1}=1})$
6:    $\kappa = \kappa + 1$
7: **while** $C(\boldsymbol{z}^{\kappa-1}, \boldsymbol{u}^{\kappa-1}) - C(\boldsymbol{z}^\kappa, \boldsymbol{u}^\kappa) > 0$

**Output:** $(\boldsymbol{z}^\kappa, \boldsymbol{u}^\kappa)$

---

## 6. Beyond Fairness: Incorporating $F_1$ Score into Binary Classification with Unbalanced Data

Unbalanced datasets for the binary classification are often refereed to those having significantly uneven instance sizes for the two classes. In an unbalanced dataset, although the majority of instances are from one class, the decision-makers are usually interested in understanding the other class with much fewer instances. Having an unbalanced dataset is common in machine learning. Many problems are inherently unbalanced, such as fraud detection (Dal Pozzolo et al., 2014), anomaly detection (Wang et al., 2016), and facial expression recognition (Rodriguez et al., 2017). The unbalanced data can influence the predictive performance considerably since most classification algorithms have been developed with the assumption that instance sizes among different classes are equal. Definitions of different performance metrics and the evaluation of the influence of unbalanced data can be found in Jeni et al. (2013).

Remarkably, in this paper, we use one of the popular performance metrics–$F_1$ score (Jeni et al., 2013) and incorporate it into the classification model to improve the training performance. Below is its formal definition.

**Definition 19** *The $F_1$ score is the harmonic mean of the precision (P) and recall (R): $F_1 = 2P \cdot R/(P + R)$, where $P = TP/(TP + FP)$ and $R = TP/(TP + FN)$ and coefficients $TP, FP, FN$ represent numbers of true positive, false positive, and false negative, respectively.*

Following the same notation from GSVM (4), we can also benefit from the binary variables $\boldsymbol{z}$ to calculate true positive (TP), false positive (FP), and false negative (FN). Since $F_1 \in [0, 1]$ and $F_1$ score higher is better, thus we will penalize $1 - F_1$ instead, which is equivalent to

$$1 - F_1 = 1 - 2\frac{P \cdot R}{P + R} = 1 - \frac{2TP}{2TP + FP + FN} = \frac{\sum_{i \in [N]}(1 - z_i)}{N + \sum_{i \in \mathcal{N}_+} z_i - \sum_{i \in \mathcal{N}_-} z_i}, \quad (34a)$$

where two sets

$$\mathcal{N}_+ = \{i \in [N] : y_i = +1\}, \mathcal{N}_- = \{i \in [N] : y_i = -1\} \quad (34b)$$

represent collections of indices of data points from each class, respectively, and $N_+ = |\mathcal{N}_+|$ is the number of data points with positive label, $N_- = |\mathcal{N}_-|$ is the number of data points with negative label.

Next, we show that our framework (1) can be adapted to incorporate $F_1$ score to the binary classification problems with unbalanced data: (i) we study GSVM (4) with unbalanced data; and (ii) we enhance deep learning with unbalanced data.

### 6.1 GSVM with Unbalanced Data

In this subsection, we will extend GSVM (4) to incorporate $F_1$ score. Penalizing $1 - F_1$ with non-negative penalty parameter $\rho$ into GSVM (4), we obtain GSVM-$F_1$

$$
\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z}} \left\{ \underbrace{\sum_{i\in\mathcal{N}_+} \frac{z_i(u_i - t)}{N_+} + \sum_{i\in\mathcal{N}_-} \frac{z_i(u_i - t)}{N_-}}_{\mathcal{A}(\boldsymbol{z},t\boldsymbol{e}-\boldsymbol{u})} + \lambda \underbrace{\|\boldsymbol{w}\|_2^2}_{\mathcal{R}(\boldsymbol{w},b,\boldsymbol{u})} + \underbrace{\frac{\rho \sum_{i\in[N]}(1 - z_i)}{N + \sum_{i\in\mathcal{N}_+} z_i - \sum_{i\in\mathcal{N}_-} z_i}}_{\mathcal{F}(\boldsymbol{z})} : (4b) - (4d) \right\},
$$
(35)

where we normalize the positive and negative instances by their sizes $N_+$ and $N_-$, respectively. This weighting strategy helps increase the prediction accuracy of the minority class, which is widely-used for unbalanced classification (Xanthopoulos and Razzaghi, 2014). Note that the proposed GSVM-$F_1$ (35) resembles framework (1), where $\mathcal{F}(\boldsymbol{z}) = 1 - F_1$. We observe that GSVM-$F_1$ (35) can be formulated as an MICP using McCormick inequalities (5) and linearizing the fractional term in the objective.

**Proposition 20** GSVM-$F_1$ (35) *can be formulated as the following MICP*

$$
\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z},\boldsymbol{s},\eta} \sum_{i\in\mathcal{N}_+} \frac{s_i - z_i t}{N_+} + \sum_{i\in\mathcal{N}_-} \frac{s_i - z_i t}{N_-} + \lambda\|\boldsymbol{w}\|_2^2 + \rho\eta,
$$
$$
s.t. \quad \eta A \geq \sum_{i\in[N]}(1 - z_i)^2, A = N + \sum_{i\in\mathcal{N}_+} z_i - \sum_{i\in\mathcal{N}_-} z_i,
$$
(36)
$$
(4b) - (4d), (5).
$$

**Proof** First of all, linearizing the bilinear terms $\{z_i u_i\}_{i\in[N]}$ using McCormick inequalities, we have

$$
\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z},\boldsymbol{s}} \left\{ \sum_{i\in\mathcal{N}_+} \frac{s_i - z_i t}{N_+} + \sum_{i\in\mathcal{N}_-} \frac{s_i - z_i t}{N_-} + \lambda\|\boldsymbol{w}\|_2^2 + \frac{\rho \sum_{i\in[N]}(1 - z_i)}{N + \sum_{i\in\mathcal{N}_+} z_i - \sum_{i\in\mathcal{N}_-} z_i} : (4b) - (4d), (5) \right\}.
$$

Let us define $A = N + \sum_{i\in\mathcal{N}_+} z_i - \sum_{i\in\mathcal{N}_-} z_i$. Then $\rho(1 - F_1)$ in the objective function of GSVM-$F_1$ (35) can be equivalently represented by $\rho\eta$ such that $\eta \geq \sum_{i\in[N]}(1 - z_i)/A$.

Thus, we have the following equivalent formulation of GSVM-$F_1$ as

$$\min_{\boldsymbol{w},b,\boldsymbol{u},\boldsymbol{z},\boldsymbol{s},\eta} \sum_{i\in\mathcal{N}_+} \frac{s_i - z_i t}{N_+} + \sum_{i\in\mathcal{N}_-} \frac{s_i - z_i t}{N_-} + \lambda\|\boldsymbol{w}\|_2^2 + \rho\eta,$$

$$\text{s.t.} \quad \eta \geq \sum_{i\in[N]} (1 - z_i)/A, A = N + \sum_{i\in\mathcal{N}_+} z_i - \sum_{i\in\mathcal{N}_-} z_i,$$

$$(4b) - (4d), (5).$$

According to the equivalence

$$\eta \geq \frac{\sum_{i\in[N]}(1 - z_i)}{A} = \frac{\sum_{i\in[N]}(1 - z_i)^2}{A} \quad \Leftrightarrow \quad \eta A \geq \sum_{i\in[N]}(1 - z_i)^2,$$

we arrive at (36). ∎

For the large-scale instances, we propose to solve GSVM-$F_1$ (35) using IRS Algorithm 2, where Step 4 requires a different subdata selection algorithm other than Algorithm 1. In fact, we can adapt Algorithm 1 to solve GSVM-$F_1$ (35) with time complexity $O(N \log N)$ when the continuous variables $(\boldsymbol{w}, b, \boldsymbol{u})$ are fixed.

**Proposition 21** *For any fixed $(\boldsymbol{w}, b, \boldsymbol{u})$ in GSVM-$F_1$ (35), optimizing over $\boldsymbol{z}$ can be done in time complexity $O(N \log N)$.*

**Proof** We split the proof into five steps.

**Step 1.** Suppose that $\sum_{i\in\mathcal{N}_-} z_i = k_-$ for any integer $k_- \in \{0, 1, \ldots, N_-\}$. Then the restricted GSVM-$F_1$ (35) is equivalent to solving the following two optimization problem

$$\{z_i^*\}_{i\in\mathcal{N}_+} \in \arg\min_{\{z_i\}_{i\in\mathcal{N}_+}} \left\{ \sum_{i\in\mathcal{N}_+} \frac{z_i(u_i - t)}{N_+} + \frac{\rho\left(N - \sum_{i\in\mathcal{N}_+} z_i - k_-\right)}{N + \sum_{i\in\mathcal{N}_+} z_i - k_-} \right\}, \tag{37a}$$

$$\{z_i^*\}_{i\in\mathcal{N}_-} \in \min_{\{z_i\}_{i\in\mathcal{N}_-}} \left\{ \sum_{i\in\mathcal{N}_-} \frac{z_i(u_i - t)}{N_-} : \sum_{i\in\mathcal{N}_-} z_i = k_- \right\}, \tag{37b}$$

where both problems can be solved efficiently via sorting two lists $\{\widehat{u}_i := (u_i - t)/N_+\}_{i\in\mathcal{N}_+}$ and $\{\widehat{u}_i := (u_i - t)/N_-\}_{i\in\mathcal{N}_-}$ in the ascending order. Namely, we suppose that $\{\widehat{u}_i\}_{i\in\mathcal{N}_+}$ and $\{\widehat{u}_i\}_{i\in\mathcal{N}_-}$ are sorted as $\widehat{u}_{\sigma_+(1)} \leq \ldots \leq \widehat{u}_{\sigma_+(N_+)}$, $\widehat{u}_{\sigma_-(1)} \leq \ldots \leq \widehat{u}_{\sigma_-(N_-)}$, respectively.

**Step 2.** Observe that the optimization problem (37b) can be solved by letting $z_{\sigma_-(\ell)}^* = 1$ for each $\ell \in [k_-]$, and 0 for each $\ell \in [N_-] \setminus [k_-]$.

**Step 3.** Next, let $\tau^* = \arg\max_{\ell\in[N_+]}\{\widehat{u}_{\sigma_+(\ell)} < 0\}$, where we let $\tau^* = 0$ if the maximizer does not exist. In the subproblem (37a), the first term is linear in $\boldsymbol{z}$ and is thus non-increasing in $\{z_{\sigma_+(\ell)}\}_{\ell\in[\tau^*]}$ and is non-decreasing in $\{z_{\sigma_+(\ell)}\}_{\ell\in[N_+]\setminus[\tau^*]}$, the second term is convex and non-increasing over the summation $\sum_{i\in\mathcal{N}_+} z_i := k_+$. Thus, according to monotonicity, we must have $z_{\sigma_+(\ell)}^* = 1$ for each $\ell \in [\tau^*]$.

Next, we can use bisection approach to search the best $k_+ \in \{\tau^*, \tau^* + 1, \ldots, N_+\}$ such that the objective is minimized, denoted by $k_+^*$, i.e., the largest $k_+ \geq \tau^*$ such that

$$\widehat{u}_{\sigma_+(k_+)} + \frac{\rho(N - k_+ - k_-)}{N + k_+ - k_-} - \frac{\rho(N - (k_+ - 1) - k_-)}{N + (k_+ - 1) - k_-} \leq 0.$$

Now let $z^*_{\sigma_+(\ell)} = 1$ if $\ell \in [k^*_+]$ and 0 for each $\ell \in [N_+] \setminus [k^*_+]$.

**Step 4.** Combining Steps 2 and 3, we have $\{z^*_i\}_{i \in \mathcal{N}_+}, \{z^*_i\}_{i \in \mathcal{N}_-}$ solve the subproblems (37a) and (37b), respectively. Now we choose the best $\boldsymbol{z}^*$ which achieves the smallest objective value among the possible ones for all $k_- \in \{0, 1, \dots, D_-\}$.

**Step 5.** Note that the sortings of two lists $\{\widehat{u}_i := (u_i - t)/N_+\}_{i \in \mathcal{N}_+}$ and $\{\widehat{u}_i := (u_i - t)/N_-\}_{i \in \mathcal{N}_-}$ can be done beforehand, and the running time complexity for each bisection is $O(\log(N_+))$. Thus, the overall running time complexity is $O(N \log N)$. ∎

The detailed implementation can be found in Algorithm 6. We remark that (i) GSVM-$F_1$ can be generalized to other binary classification models with $F_1$ score to measure the learning outcomes; and (ii) GSVM-$F_1$ can be solved by the proposed IRS Algorithm 2 when replacing Step 4 with employing Algorithm 6 to obtain $\boldsymbol{z}^{t+1}$.

---

**Algorithm 6** Algorithm for solving GSVM-$F_1$ with fixed $(\boldsymbol{w}, b, \boldsymbol{u})$

---

**Input:** $\{\boldsymbol{x}_i, y_i\}_{i \in [N]}, (\boldsymbol{w}, b, \boldsymbol{u}), \rho > 0, \lambda > 0$

1: Initialize $k_- = 0$, $k_+ = 0$, and $\mathcal{N}_+ = \{i \in [N] : y_i = +1\}$, $\mathcal{N}_- = \{i \in [N] : y_i = -1\}$ with $N_+ = |\mathcal{N}_+|$, $N_- = |\mathcal{N}_-|$

2: Sort $\{\widehat{u}_i = (u_i - t)/N_+\}_{i \in \mathcal{N}_+}$ and $\{\widehat{u}_i = (u_i - t)/N_-\}_{i \in \mathcal{N}_-}$ in the ascending order such that $\widehat{u}_{\sigma_+(1)} \leq \dots \leq \widehat{u}_{\sigma_+(N_+)}$, $\widehat{u}_{\sigma_-(1)} \leq \dots \leq \widehat{u}_{\sigma_-(N_-)}$, respectively

3: Let $\tau^* = \arg\max_{\ell \in [N_+]}\{\widehat{u}_{\sigma_+(\ell)} < 0\}$ and let $\tau^* = 0$ if the maximizer does not exist

4: Calculate $\left\{\widehat{U}_{\sigma_+}(k_{b+}) := \sum_{i \in [k_{b+}]} \widehat{u}_{\sigma_+(i)}\right\}_{k_{b+} \in [N_+] \setminus [\tau^* - 1]}$ and $\left\{\widehat{U}_{\sigma_-}(k_-) := \sum_{i \in [k_-]} \widehat{u}_{\sigma_-(i)}\right\}_{k_- \in [N_-]}$

5: **while** $k_- \leq N_-$ **do**

6:     Let $\underline{m} = m$, $k_{b0} = \tau^*$ and $k_{b1} = N_+$

7:     **while** $\tau^* \geq 1$ and $k_{b1} - k_{b0} \geq 1$ **do**

8:         $k_{b+} = \lfloor (k_{b0} + k_{b1})/2 \rfloor$

9:         $\Delta = \widehat{u}_{\sigma_+(k_{b+})} + \frac{\rho(N - k_{b+} - k_-)}{N + k_{b+} - k_-} - \frac{\rho(N - (k_{b+} - 1) - k_-)}{N + (k_{b+} - 1) - k_-}$

10:         **if** $\Delta \leq 0$ **then**

11:             $k_{b0} := k_{b+}$

12:         **else**

13:             $k_{b1} := k_{b+}$

14:     Let $v_{k_-} = \widehat{U}_{\sigma_+}(k_{b+}) + \widehat{U}_{\sigma_-}(k_-) + \rho(N - k_{b+} - k_-)/(N + k_{b+} - k_-)$

15:     Let $k_+(k_-) = k_{b+}$ and $k_- = k_- + 1$

16: Let $k^*_- = \arg\min_{k_-=0,\dots,N_-}\{v_{k_-}\}, k^*_+ = k_+(k^*_-), v^* = v_{k^*_-}, z^*_i = 1$ if $i \in \{\sigma_+(\ell)\}_{\ell \in [k^*_+]} \cup \{\sigma_-(\ell)\}_{\ell \in [k^*_-]}$, and 0, otherwise

**Output:** $(\boldsymbol{z}^*, v^*)$

---

## 6.2 Deep Learning with Unbalanced Data

Similar to GSVM-$F_1$ (35), we propose the following formulation for CNN with $F_1$ score (CNN-$F_1$)

$$\min_{\boldsymbol{z} \in \{0,1\}^N} \left\{ \underbrace{\sum_{i \in \mathcal{N}_+} \frac{z_i (t - u_i)}{N_+} + \sum_{i \in \mathcal{N}_-} \frac{z_i (t - u_i)}{N_-}}_{\mathcal{A}(\boldsymbol{z}, t\boldsymbol{e} - \boldsymbol{u})} + \underbrace{\frac{\rho \sum_{i \in [N]} (1 - z_i)}{N + \sum_{i \in \mathcal{N}_+} z_i - \sum_{i \in \mathcal{N}_-} z_i}}_{\mathcal{F}(\boldsymbol{z})} \right\}, \tag{38}$$

where we normalize the positive and negative instances by their sizes $N_+$ and $N_-$, respectively, and $\boldsymbol{u}$ is the solution of VCNN with each entry $u_i$ denoting the probability of correctly predicting the label $y_i$ for each $i \in [N]$. We see that the proposed CNN-$F_1$ (38) can be viewed as a special case of the framework (1), where $\mathcal{R}(\boldsymbol{w}, b, \boldsymbol{u}) = 0$ and $\mathcal{F}(\boldsymbol{z}) = 1 - F_1$.

Note that (i) the proposed IRS Algorithm 5 can be used to solve CNN-$F_1$ , where we replace Step 4 by employing Algorithm 6 to obtain $\boldsymbol{z}^{\kappa+1}$ and in Algorithm 6, and we redefine $\{\widehat{u}_i = (t - u_i)/N_+\}_{i \in \mathcal{N}_+}$ and $\{\widehat{u}_i = (t - u_i)/N_-\}_{i \in \mathcal{N}_-}$; and (ii) CNN-$F_1$ can be generalized to other deep learning models with $F_1$ score to measure the quality of classification outcomes.

## 7. Numerical Experiments

We conduct numerical studies to: (a) demonstrate the performance of Algorithm 1 compared with Gurobi solver, (b) demonstrate the performance of IRS Algorithm 2 compared with Gurobi solver, (c) test whether GSVMF (12) can indeed improve classification fairness, (d) compare the performance of GSVMF, GKSVMF, and GLRF, (e) illustrate the performance of GSVMF by comparing our IRS Algorithm 2 for solving GSVMF (26) with existing SSVM method proposed by Olfat and Aswani (2017), and (f) illustrate the performance of fair deep learning and deep learning with unbalanced data. All the instances in this section were executed in Python 3.7 with calls to solver Gurobi (version 9.0 with default settings) on a personal PC with 2.3 GHz Intel Core i9 processor and 16G of memory. Codes of the numerical experiments are available at `https://github.com/qingye1/Fair_Classification`.

## 7.1 Algorithm 1 Testing

**Experiment 1:** In this experiment, we conducted a comparison between the proposed Algorithm 1 and Gurobi solver to demonstrate the effectiveness of Algorithm 1. We compared the running time for solving GSVMF (12) with various problem sizes $N$. We varied $N$ over 50, 100, 200, 500, 1000, 1500, 2000, and 5000. We generated each instance by truncating the wine quality (WQ) dataset (Dua and Graff, 2017) into smaller sizes. This dataset contains 12 features and wine quality scores. We labeled $y_i = +1$ if a wine has a score of 6 or higher and $y_i = -1$, otherwise. We defined $g_i = +1$ for white wines and $g_i = -1$ for red wines as the sensitive feature. We set $b = 100$, generated $\boldsymbol{w}$ with each entry being uniformly distributed in the interval $[-10.0, 10.0]$ and generated $\boldsymbol{u}$ with each entry being uniformly distributed in the interval $[0, 10.0]$. For each instance, we set $t = 1$, varied $\rho$ over 0.01, 0.03,

0.05, 0.1, 0.2, 0.5, 0.8, 1.0, 2.0, 3.0, 5.0, and 10.0, and output the average running time in seconds. Note that we set the time limit of Gurobi to be 600 seconds.

Figure 3 illustrates the average running time. We see that Algorithm 1 is always faster than Gurobi. In the first five tests, both Algorithm 1 and Gurobi solve the program within 3 seconds. Starting from the size $N = 1500$, Gurobi spends a considerably long time on solving GSVMF (12). Since we set the time limit to be 600 seconds, the output time is shorter than the actual time required by Gurobi. Nevertheless, the running time for Algorithm 1 grows much slower than that of Gurobi in the figure. This demonstrates that the proposed Algorithm 1 significantly outperforms Gurobi.
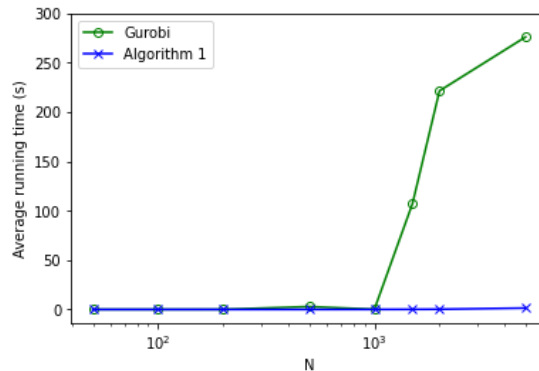


Figure 3: Average Running Time of Gurobi and Algorithm 1 with Different $N$ in Experiment 1. The plot is shown on a logarithmic scale.

### 7.2 IRS Algorithm 2 Testing

**Experiment 2:** In this experiment, we conducted a comparison between IRS Algorithm 2 and Gurobi solver. We compared the objective values and running time for solving GSVMF (12) with different $\rho$ and we ran Gurobi to solve MICP (6). We generated the instance by extracting 55 data points from the same WQ dataset in Experiment 1. We set $\lambda = 1, t = 1$, and varied $\rho$ over 0.01, 0.03, 0.05, 0.1, 0.2, 0.5, 0.8, 1.0, 2.0, 3.0, 5.0, and 10.0.

The running time and objective values are displayed in Table 4, where opt.val denotes the optimal value of Gurobi and obj.val denotes the objective value of IRS Algorithm 2. For these cases, Gurobi can solve (6) to optimality, and we thus computed the relative optimality gap of IRS Algorithm 2, denoted by Gap. The running time for the proposed Algorithm 2 is 0.03 seconds for most cases, which is very stable. Gurobi takes a much longer time than IRS. Its running time varies from 9 seconds to 84 seconds, even with $N = 55$. The optimality gaps are within 2.5% for all cases. This demonstrates that IRS can consistently and effectively solve GSVMF (12) to near-optimality.

### 7.3 Subdata Selection Testing

**Experiment 3:** In this experiment, we examined the ability of the proposed GSVMF (12) to improve the fairness by comparing with SVM (2). We generated 200 data points with two dimensional features $(x_1, x_2)$ besides the sensitive feature $g$. The two dimensional features for the data points in different classes and groups were generated from normal distribution with different parameters. Particularly, for the first set of 50 data points with label $y_i = +1$ and sensitive feature $g_i = +1$, their features $(x_1, x_2)$ were generated according to a two-dimensional normal distribution with mean $(3, 4)$ and covariance matrix $\begin{bmatrix} 4 & 0 \\ 0 & 9 \end{bmatrix}$. For the second set of 50 data points with label $y_i = +1$ and sensitive feature $g_i = -1$, their features $(x_1, x_2)$ were generated according to a two-dimensional normal distribution with

| $N = 55, \lambda = 1, t = 1$ | Gurobi | | Proposed IRS | | |
|---|---|---|---|---|---|
| $\rho$ | opt.val | Time (s) | obj.val | Gap (%) | Time (s) |
| 0.01 | -0.6852 | 14 | -0.6827 | 0.4 | 0.03 |
| 0.03 | -0.6829 | 14 | -0.6804 | 0.4 | 0.03 |
| 0.05 | -0.6805 | 13 | -0.6781 | 0.4 | 0.02 |
| 0.1 | -0.6747 | 14 | -0.6722 | 0.4 | 0.02 |
| 0.2 | -0.6631 | 12 | -0.6606 | 0.4 | 0.02 |
| 0.5 | -0.6387 | 9 | -0.6257 | 2.0 | 0.02 |
| 0.8 | -0.6328 | 16 | -0.6275 | 0.8 | 0.03 |
| 1.0 | -0.6309 | 14 | -0.6150 | 2.5 | 0.02 |
| 2.0 | -0.6211 | 29 | -0.6089 | 2.0 | 0.03 |
| 3.0 | -0.6140 | 42 | -0.6075 | 1.1 | 0.03 |
| 5.0 | -0.6112 | 45 | -0.6047 | 1.1 | 0.03 |
| 10.0 | -0.6042 | 84 | -0.5977 | 1.1 | 0.04 |

Table 4: Comparison between Gurobi and the Proposed IRS in Experiment 2

mean $(2, 6)$ and covariance matrix $\begin{bmatrix} 4 & 0 \\ 0 & 9 \end{bmatrix}$. For the third set of 50 data points with label $y_i = -1$ and sensitive feature $g_i = +1$, their features $(x_1, x_2)$ were generated according to a two-dimensional normal distribution with mean $(7, 5)$ and covariance matrix $\begin{bmatrix} 4 & 0 \\ 0 & 9 \end{bmatrix}$. For the fourth set of 50 data points with label $y_i = -1$ and sensitive feature $g_i = -1$, their features $(x_1, x_2)$ were generated according to a two-dimensional normal distribution with mean $(8, 3)$ and covariance matrix $\begin{bmatrix} 4 & 0 \\ 0 & 9 \end{bmatrix}$.

The SVM results were obtained by solving (2) with the best tuned $\lambda = 0.5$. Different values of tuning parameters $(t, \lambda, \rho)$ of GSVMF (12) were used to optimize the accuracy and fairness level. Particularly, we varied $t$ over $0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 1.5$, and $2.0$. For $\lambda$, we tried $0, 1/(1000N), 1/(100N), 1/(2N), 1/N, 2/N, 10/N, 100/N$, and $1000/N$. We varied $\rho$ over $0.01, 0.1, 0.2, 0.5, 0.8, 1.0, 2.0, 3.0, 5.0, 10.0$, and $20.0$. The numerical results are illustrated in Figure 4.

Figure 4a plots the 200 data points and demonstrates the decision boundary for SVM. The accuracy for SVM is 90% and the OMR fairness is 14%. Figure 4b illustrates the result of GSVMF, where the best tuning parameters values $(t, \lambda, \rho)$ were selected as $(0.3, 0, 0.2)$. The decision boundary of the proposed GSVMF is to counterclockwise rotates that of SVM by around 30° to improve the fairness from 14% to 0% with only 2% decrease of accuracy. The data points represented by the filled markers in Figure 4b are the biased ones dropped by the subdata selection method in order to generate a much more fair decision boundary. This demonstrates that incorporating the subdata selection method can largely improve classification fairness.

## 7.4 Comparisons of Different Classification Models

**Experiment 4 (Comparing Different Binary Classification Methods with OMR Fairness):** In this experiment, we conducted a thorough comparison between GSVMF

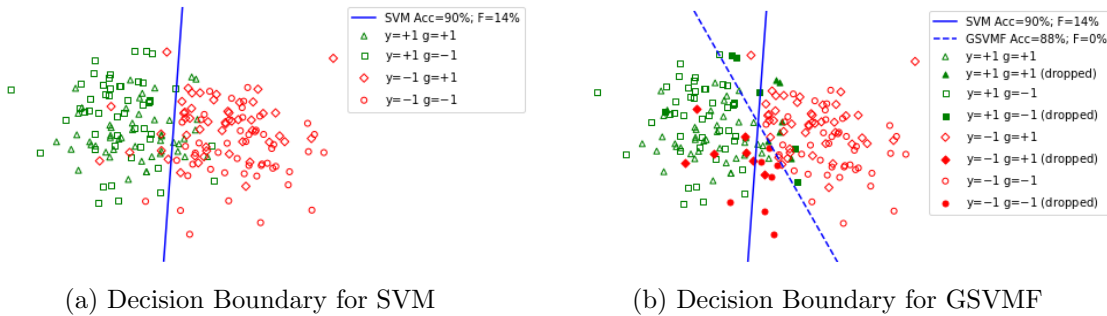(a) Decision Boundary for SVM  (b) Decision Boundary for GSVMF

Figure 4: A Comparison of SVM and the Proposed GSVMF in Experiment 3.

(12), GKSVMF (32), and GLRF (31) with OMR fairness in Definition 6 using ProPublica COMPAS dataset (Jeff Larson and Angwin) and four real datasets from UCI (Dua and Graff, 2017).

We preprocessed the data using one-hot encoding and normalization. For COMPAS (CP) dataset, we used the same features as Jeff Larson and Angwin. We labeled $y_i = +1$ if a person would recidivate within two years and $y_i = -1$, otherwise. We defined $g_i = +1$ for Black and $g_i = -1$ for White. We set gender as the sensitive feature for all the four UCI datasets. Specifically, we defined $g_i = +1$ for male and $g_i = -1$ for female. For default payment (DP) dataset, we labeled $y_i = +1$ for default payment and $y_i = -1$, otherwise. For abalone (AB) dataset, we labeled $y_i = +1$ if the abalone is above the age of 10 years old and $y_i = -1$, otherwise. For Portuguese grade (PG) dataset and math grade (MG) dataset, we labeled $y_i = +1$ for grade above the median and $y_i = -1$, otherwise.

Different values of tuning parameters $(t, \lambda, \rho)$ of GSVMF (12), GLRF (31), and $(t, \rho, kernel)$ of GKSVMF (32) were used to obtain the best accuracy and fairness level. Particularly, we varied t over 0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 1.5, and 2.0 for GSVMF (12) and GKSVMF (32), and $\log(0.1), \log(0.3), \log(0.5), \log(0.7), \log(0.9), \log(1.0), \log(1.5)$, and $\log(2.0)$ for GLRF (31). For $\lambda$, we tried 0, 1/(1000N), 1/(100N), 1/(2N), 1/N, 2/N, 10/N, 100/N, and 1000/N. We varied $\rho$ over 0.01, 0.1, 0.2, 0.5, 0.8, 1.0, 2.0, 3.0, 5.0, 10.0, and 20.0. For GKSVMF (12), we used RBF and poly kernels.

We compared GSVMF (12), GKSVMF (32), and GLRF (31) based on the prediction accuracy (Acc) and fairness (F). Smaller fairness measure (F) is better. The best scenario was selected by the trade-off ratio Acc/F, where a large ratio represents a better trade-off between accuracy and fairness. We used 5-fold cross-validation with a 70/30 training and testing split for all the datasets. GSVMF (12), GKSVMF (32), and GLRF (31) were solved using the proposed IRS Algorithm 2, where we used gradient descent method to solve $(\boldsymbol{w}, b, \boldsymbol{u})$ for GSVMF (12) and GLRF (31), while GKSVMF (32) was solved by Algorithm 4 with a call of python package sklearn.svm.SVC.

The results displayed in Table 5 are those with the largest Acc/F ratio for each dataset. Their corresponding parameters, accuracy, fairness, and training time are shown in the table. It is seen that GSVMF, GKSVMF, and GLRF have similar prediction accuracy for each dataset. For CP, AB, PG, and MG datasets, GSVMF has the best Acc/F ratio. For DP and PG datasets, GKSVMF has the best Acc/F ratio. Overall, GKSVMF has shorter training time. Therefore, in practice, we recommend running different models to choose the one having the best fairness given the desired accuracy requirement.

34

| Dataset | Features | $N$ | Methods | Parameters | Testing | | | Time (s) |
|---------|----------|-----|---------|------------|---------|---|---|----------|
| | | | | | $Acc$ (%) | $F$ (%) | $Acc/F$ | |
| DP | 23 | 30000 | GSVMF | $(0.3, 1/N, 0)$[i] | 80.5 | 1.8 | 45 | 37.8 |
| | | | GKSVMF | $(0.5, 0.8, \text{poly})$[ii] | 79.6 | 0.1 | 796 | 30.3 |
| | | | GLRF | $(\log(0.5), 0, 0)$[iii] | 81.5 | 1.7 | 48 | 33.1 |
| CP | 5 | 5278 | GSVMF | $(2, 1/N, 0)$ | 66.0 | 0.02 | 3300 | 2.5 |
| | | | GKSVMF | $(0.7, 2, \text{RBF})$ | 64.0 | 0.03 | 2133 | 1.9 |
| | | | GLRF | $(\log(0.3), 0, 0)$ | 66.4 | 0.06 | 1107 | 2.5 |
| AB | 8 | 2835 | GSVMF | $(0.5, 1/(1000N), 0.8)$ | 71.8 | 0.2 | 359 | 2.2 |
| | | | GKSVMF | $(0.1, 20, \text{poly})$ | 68.6 | 0.3 | 229 | 1.2 |
| | | | GLRF | $(\log(0.1), 10/N, 0)$ | 70.4 | 2.2 | 32 | 1.6 |
| PG | 32 | 649 | GSVMF | $(0.5, 10/N, 0)$ | 93.3 | 0.1 | 933 | 0.7 |
| | | | GKSVMF | $(0.5, 0, \text{poly})$ | 93.3 | 0.1 | 933 | 0.2 |
| | | | GLRF | $(\log(0.1), 100/N, 0)$ | 94.9 | 0.2 | 475 | 2.6 |
| MG | 31 | 395 | GSVMF | $(0.7, 10/N, 0)$ | 93.3 | 0.1 | 933 | 0.4 |
| | | | GKSVMF | $(2, 0, \text{RBF})$ | 92.4 | 1.8 | 51 | 0.1 |
| | | | GLRF | $(\log(0.5), 100/N, 0)$ | 91.6 | 0.1 | 916 | 0.8 |

[i] The 3-tuple represents $(t, \lambda, \rho)$;
[ii] The 3-tuple represents $(t, \rho, \text{kernel})$;
[iii] The 3-tuple represents $(t, \lambda, \rho)$.

Table 5: Performance of GSVMF, GKSVMF, and GLRF with OMR Fairness in Experiment 4

**Experiment 5 (Comparing GSVMF** (26) **and SSVM in Olfat and Aswani (2017) with DP Fairness):** In this experiment, we conducted a comparison between our proposed GSVMF (26) and SSVM proposed by Olfat and Aswani (2017) with DP fairness in Section 4.3 using the same datasets as Experiment 4.

Different values of tuning parameters $(t, \lambda, \rho)$ of GSVMF (26) and $(d, \mu)$ of SSVM were used to obtain the best accuracy and fairness level. Particularly, we varied t over 0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 1.5, and 2.0. For $\lambda$, we tried 0, 1/(1000N), 1/(100N), 1/(2N), 1/N, 2/N, 10/N, 100/N, and 1000/N. We varied $\rho$ over 0.01, 0.1, 0.2, 0.5, 0.8, 1.0, 2.0, 3.0, 5.0, 10.0, and 20.0. The values of $(d, \mu)$ were selected to be the same as Olfat and Aswani (2017).

Similar to Experiment 4, we compared GSVMF (26) and SSVM based on the prediction accuracy (Acc) and fairness (F). The SSVM results were computed using their implementation, which is available at `https://github.com/molfat66/FairML`. Both models were tuned using 5-fold cross-validation.

The results displayed in Table 6 are those with the largest trade-off ratio for each dataset. The corresponding parameters, accuracy, fairness, and training time are shown in the table. In all the datasets, GSVMF has a considerably larger ratio and shorter training time than SSVM. For some datasets, GSVMF has a slightly lower accuracy than SSVM due to our selection criterion. Nevertheless, the proposed GSVMF's fairness is significantly better than SSVM. Thus, our approach always has a larger Acc/F ratio. Besides, GSVMF can reduce fairness to less than 1% with good accuracy for the first three datasets. In terms of training time, we see that both methods take a similar amount of time. Therefore, we conclude

| Dataset | Features | $N$ | Methods | Parameters | Testing | | | Time (s) |
|---|---|---|---|---|---|---|---|---|
| | | | | | $Acc$ (%) | $F$ (%) | $Acc/F$ | |
| DP | 23 | 30000 | SSVM | $(0, 0.01)^{\text{i}}$ | 71.7 | 2.8 | 26 | 48.5 |
| | | | GSVMF | $(2, 1/(100N), 0.01)^{\text{ii}}$ | 82.5 | 0.6 | 138 | 39.4 |
| CP | 5 | 5278 | SSVM | $(0.1, 0.01)$ | 58.0 | 9.8 | 6 | 20.7 |
| | | | GSVMF | $(1.5, 1/(2N), 0.5)$ | 64.9 | 0.8 | 81 | 2.9 |
| AB | 8 | 2835 | SSVM | $(0.1, 0.01)$ | 73.4 | 3.4 | 22 | 21.8 |
| | | | GSVMF | $(0.5, 1/(1000N), 3)$ | 71.8 | 0.08 | 898 | 2.4 |
| PG | 32 | 649 | SSVM | $(0.1, 1.0)$ | 96.7 | 15.9 | 6 | 14.8 |
| | | | GSVMF | $(0.9, 10/N, 0.2)$ | 94.9 | 3.6 | 26 | 0.8 |
| MG | 31 | 395 | SSVM | $(0.05, 0.03)$ | 96.1 | 9.0 | 11 | 17.6 |
| | | | GSVMF | $(0.5, 10/N, 0)$ | 94.1 | 4.3 | 22 | 0.5 |

[i] The 2-tuple represents $(d, \mu)$;
[ii] The 3-tuple represents $(t, \lambda, \rho)$.

Table 6: Performance of SSVM and GSVMF with DP Fairness in Experiment 5

that using binary variables to incorporate exact fairness measure can indeed significantly improve the fairness.

## 7.5 A Comparison between VCNN and FCNN

**Experiment 6:** In this experiment, we conducted a comparison between VCNN (i.e., vanilla CNN) and FCNN (33) using Algorithm 5. The datasets used in this experiment were age dataset (Cheng et al., 2019), race dataset (Cheng et al., 2019), gender dataset (Zhang et al., 2017), and X-ray dataset (xra, 2017). For age dataset, we labeled $y = +1$ for 20s and $y = -1$ for 60s with race of Asian or White as sensitive feature. For race dataset, we labeled $y = +1$ for Asian and $y = -1$ for White with age of 20s or 60s as sensitive feature. For gender dataset, we labeled $y = +1$ for male and $y = -1$ for female with age of 10s or 70s as sensitive feature. For chest X-ray datasets named infiltration and atelectasis, we labeled $y = +1$ for infiltration and atelectasis disease and $y = -1$ for normal with gender of male or female as sensitive feature, respectively. We resized the image data to $50 \times 50$ pixels and converted to black and white. For both VCNN and FCNN (33), we chose 20 for epoch, adam for optimizer, and binary cross entropy for loss function and performed training for kernel size (ks) of $3 \times 3$ and $5 \times 5$. For FCNN (33), we varied t over 0.1, 0.3, 0.5, 0.7, 0.9, 0.98, and varied $\rho$ over $0, 1/(10N), 1/(2N), 1/N, 5/N, 10/N$. In IRS Algorithm 5, we solved VCNN using Keras Sequential model in Python deep learning library and then performed IRS procedure for four iterations for the sake of time. The scenario with the best fairness from the four iterations was selected as the output. We ran the IRS Algorithm 5 for five times and output the average and best results. We compared the testing accuracy and fairness for VCNN and FCNN (33).

Table 7 displays the testing accuracy (Acc) and fairness (F), kernel size (ks) and the best tuning parameters $(t, \rho)$, and running time for different datasets. Smaller fairness measure (F) is better. It is seen that the proposed FCNN improves both average fairness and accuracy in most scenarios. For example, our approach improves fairness from 12.5%

to 2.7% with a 4.9% improvement of accuracy for race dataset when $ks = 3 \times 3$. FCNN improves fairness from 12.0% to 1.2% with a 1.4% improvement of accuracy for X-ray infiltration dataset when $ks = 5 \times 5$. The best accuracy and fairness for FCNN also outperform VCNN in most datasets. We also observe that due to the iterative procedure in IRS Algorithm 5, the training time for our method is longer than that of VCNN. Overall, the proposed FCNN indeed outperforms VCNN both in accuracy and fairness.

| Dataset | $N$ | Methods | VCNN | FCNN | VCNN | FCNN | VCNN | FCNN | VCNN | FCNN |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Average | | Best | | Average | | Best | |
| Age | 405 | Parameters | ks=3 × 3, t=0.9, $\rho$=1/(2N) | | | | ks=5 × 5, t=0.9, $\rho$=1/(2N) | | | |
| | | Testing $Acc(\%)$ | 60.2 | 63.9 | 61.5 | 67.2 | 57.2 | 63.6 | 61.5 | 65.6 |
| | | Testing $F(\%)$ | 5.8 | 0.3 | 2.3 | 0.1 | 4.5 | 0.3 | 1.1 | 0.1 |
| | | Time (s) | 21 | 82 | 21 | 78 | 26 | 90 | 25 | 84 |
| Race | 405 | Parameters | ks=3 × 3, t=0.9, $\rho$=1/(10N) | | | | ks=5 × 5, t=0.98, $\rho$=5/N | | | |
| | | Testing $Acc(\%)$ | 53.8 | 58.7 | 54.9 | 61.5 | 54.9 | 56.1 | 57.4 | 61.5 |
| | | Testing $F(\%)$ | 12.5 | 2.7 | 6.9 | 1.1 | 10.3 | 2.2 | 6.9 | 0.3 |
| | | Time (s) | 34 | 113 | 31 | 101 | 45 | 132 | 41 | 103 |
| Gender | 629 | Parameters | ks=3 × 3, t=0.98, $\rho$=10/N | | | | ks=5 × 5, t=0.98, $\rho$=1/N | | | |
| | | Testing $Acc(\%)$ | 66.7 | 67.1 | 68.1 | 69.6 | 69.2 | 67.7 | 72.3 | 70.2 |
| | | Testing $F(\%)$ | 3.5 | 0.6 | 1.2 | 0.1 | 5.1 | 0.6 | 1.2 | 0.1 |
| | | Time (s) | 33 | 122 | 33 | 113 | 40 | 127 | 39 | 118 |
| X-ray Infiltration | 368 | Parameters | ks=3 × 3, t=0.5, $\rho$=1/(2N) | | | | ks=5 × 5, t=0.5, $\rho$=1/(10N) | | | |
| | | Testing $Acc(\%)$ | 67.6 | 67.6 | 75.7 | 72.1 | 64.5 | 65.9 | 69.4 | 71.2 |
| | | Testing $F(\%)$ | 9.8 | 0.8 | 6.1 | 0.5 | 12.0 | 1.2 | 8.4 | 0.5 |
| | | Time (s) | 28 | 89 | 27 | 66 | 35 | 120 | 26 | 91 |
| X-ray Atelectasis | 1122 | Parameters | ks=3 × 3, t=0.3, $\rho$=10/N | | | | ks=5 × 5, t=0.3, $\rho$=1/(2N) | | | |
| | | Testing $Acc(\%)$ | 61.7 | 65.6 | 64.7 | 67.1 | 63.1 | 63.4 | 65.6 | 66.2 |
| | | Testing $F(\%)$ | 2.7 | 0.4 | 1.5 | 0.2 | 3.4 | 0.6 | 1.4 | 0.2 |
| | | Time (s) | 36 | 129 | 34 | 119 | 43 | 143 | 42 | 139 |

Table 7: Performance of FCNN with OMR Fairness in Experiment 6

## 7.6 Deep Learning with Unbalanced Datasets

**Experiment 7:** In this experiment, we conducted a comparison between VCNN (i.e., vanilla CNN) and CNN-$F_1$ (38). We used the same datasets and Keras Sequential model setting as Experiment 6. To generate unbalanced datasets, we reduced the sizes of the datasets for 60s class, Asian class, male class, infiltration class, and atelectasis class to 22% of age dataset, 22% of race dataset, 20% of gender dataset, 24% of infiltration dataset, and 15% of atelectasis dataset, respectively. We performed training for kernel size (ks) of $3 \times 3$ and $5 \times 5$ and varied $t$ over 0.1, 0.3, 0.5, 0.7, 0.9, 0.98. We also varied $\rho$ over 0, 1/(10N), 1/(2N), 1/N, 5/N, 10/N. Similar to Experiment 6, we solved VCNN using Keras Sequential model in Python deep learning library, while we performed IRS as described in Section 6.2 for four iterations when solving CNN-$F_1$ (38). The scenario with the best $F_1$ score from the four iterations was selected as the output. We ran the IRS for five times and output the

average and best results. We compared the testing accuracy and $F_1$ score for VCNN and CNN-$F_1$ (38).

Table 8 displays the testing accuracy (Acc) and $F_1$ score, kernel size (ks) and the best tuning parameters $(t, \rho)$, and running time for different datasets. Larger $F_1$ score represents better classification result for unbalanced data. It is seen that the proposed CNN-$F_1$ improves both average $F_1$ score and accuracy in most scenarios. For example, CNN-$F_1$ improves $F_1$ score from 0.29 to 0.41 with a 7.2% improvement of accuracy for race dataset when $ks = 3 \times 3$, while it improves $F_1$ score from 0.04 to 0.27 with a 0.3% improvement of accuracy for X-ray atelectasis dataset when $ks = 5 \times 5$. The best accuracy and $F_1$ score for CNN-$F_1$ also outperform VCNN in all scenarios except the atelectasis dataset when $ks = 3 \times 3$. It is worthy of mentioning that the training time for CNN-$F_1$ is often longer than that of VCNN due to IRS, however, the proposed CNN-$F_1$ can significantly improve the accuracy and $F_1$ score.

| Dataset | N | Methods | VCNN | CNN-$F_1$ | VCNN | CNN-$F_1$ | VCNN | CNN-$F_1$ | VCNN | CNN-$F_1$ |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Average | | Best | | Average | | Best | |
| Age | 274 | Parameters | ks=3 × 3, t=0.9, $\rho$=5/N | | | | ks=5 × 5, t=0.9, $\rho$=10/N | | | |
| | | Testing $Acc$(%) | 73.4 | 74.4 | 74.6 | 80.5 | 72.4 | 74.8 | 74.4 | 79.3 |
| | | Testing $F_1$ | 0.29 | 0.32 | 0.37 | 0.40 | 0.33 | 0.45 | 0.40 | 0.49 |
| | | Time (s) | 20 | 79 | 19 | 75 | 26 | 104 | 26 | 101 |
| Race | 269 | Parameters | ks=3 × 3, t=0.9, $\rho$=1/(10N) | | | | ks=5 × 5, t=0.9, $\rho$=0 | | | |
| | | Testing $Acc$(%) | 72.1 | 79.3 | 74.1 | 81.5 | 69.9 | 76.3 | 72.8 | 79.0 |
| | | Testing $F_1$ | 0.29 | 0.41 | 0.32 | 0.47 | 0.24 | 0.38 | 0.31 | 0.41 |
| | | Time (s) | 20 | 75 | 20 | 73 | 24 | 98 | 24 | 94 |
| Gender | 391 | Parameters | ks=3 × 3, t=0.9, $\rho$=0 | | | | ks=5 × 5, t=0.9, $\rho$=1/N | | | |
| | | Testing $Acc$(%) | 79.1 | 81.4 | 79.7 | 85.6 | 78.8 | 83.6 | 80.5 | 84.7 |
| | | Testing $F_1$ | 0.31 | 0.43 | 0.35 | 0.45 | 0.27 | 0.44 | 0.34 | 0.47 |
| | | Time (s) | 36 | 143 | 35 | 138 | 45 | 170 | 44 | 166 |
| X-ray Infiltration | 246 | Parameters | ks=3 × 3, t=0.3, $\rho$=0 | | | | ks=5 × 5, t=0.1, $\rho$=10/N | | | |
| | | Testing $Acc$(%) | 67.8 | 71.9 | 70.3 | 73.0 | 68.1 | 70.0 | 73.0 | 75.7 |
| | | Testing $F_1$ | 0.14 | 0.25 | 0.32 | 0.44 | 0.03 | 0.37 | 0.09 | 0.42 |
| | | Time (s) | 26 | 96 | 25 | 91 | 46 | 182 | 45 | 179 |
| X-ray Atelectasis | 653 | Parameters | ks=3 × 3, t=0.3, $\rho$=1/N | | | | ks=5 × 5, t=0.3, $\rho$=1/N | | | |
| | | Testing $Acc$(%) | 82.0 | 79.5 | 84.2 | 82.1 | 82.0 | 82.3 | 82.7 | 83.2 |
| | | Testing $F_1$ | 0.03 | 0.28 | 0.09 | 0.36 | 0.04 | 0.27 | 0.13 | 0.33 |
| | | Time (s) | 60 | 236 | 59 | 231 | 64 | 253 | 63 | 246 |

Table 8: Performance of CNN-$F_1$ with Unbalanced Data in Experiment 7. Note that larger $F_1$ score implies better classification result.

# 8. Conclusion

We introduced a unified framework for fair classification with unbiased subdata selection procedure and exact fairness representation. The proposed framework is versatile and can be adapted to many classifiers with exact fairness representation. We proposed exact mixed-

integer convex programming formulations for the moderate-sized instances, and developed a scalable iterative refining strategy, inspired by the alternating minimization approach, to solve large-scale instances effectively. The numerical study demonstrated that our approach can enhance fairness with little or no loss of prediction accuracy or even improved accuracy. We are working towards incorporating exact fairness into the machine learning models with continuous responses such as fair regression.

## Acknowledgment

## References

NIH Chest X-rays. https://www.kaggle.com/nih-chest-xrays/data?select=Data_Entry_2017.csv, 2017.

Vahid Abolghasemi, Saideh Ferdowsi, and Saeid Sanei. A gradient-based alternating minimization approach for optimization of the measurement matrix in compressive sensing. *Signal Processing*, 92(4):999–1009, 2012.

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. *arXiv preprint arXiv:1803.02453*, 2018.

Sina Aghaei, Mohammad Javad Azizi, and Phebe Vayanos. Learning optimal and fair decision trees for non-discriminative decision-making. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1418–1426, 2019.

Lulseged Ayalew and Hiromitsu Yamagishi. The application of gis-based logistic regression for landslide susceptibility mapping in the kakuda-yahiko mountains, central japan. *Geomorphology*, 65(1-2):15–31, 2005.

Ananth Balashankar, Alyssa Lees, Chris Welty, and Lakshminarayanan Subramanian. What is fair? exploring pareto-efficiency for fairness constrained classifiers. *arXiv preprint arXiv:1910.14120*, 2019.

Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*, pages 3992–4001, 2017.

Shutong Chen and Weijun Xie. On the cluster-aware supervised learning (clusl): Frameworks, convergent algorithms, and applications. *INFORMS Journal On Computing. Accepted*, 2020.

Jingchun Cheng, Yali Li, Jilong Wang, Le Yu, and Shengjin Wang. Exploiting effective facial patches for robust gender recognition. *Tsinghua Science and Technology*, 24(3):333–345, 2019.

Andrea Dal Pozzolo, Olivier Caelen, Yann-Ael Le Borgne, Serge Waterschoot, and Gianluca Bontempi. Learned lessons in credit card fraud detection from a practitioner perspective. *Expert systems with applications*, 41(10):4915–4928, 2014.

Marleen De Bruijne. Machine learning approaches in medical image analysis: From detection to diagnosis, 2016.

Michele Donini, Luca Oneto, Shai Ben-David, John S Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. In *Advances in Neural Information Processing Systems*, pages 2791–2801, 2018.

Stephan Dreiseitl and Lucila Ohno-Machado. Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*, 35 (5-6):352–359, 2002.

Mengnan Du, Fan Yang, Na Zou, and Xia Hu. Fairness in deep learning: A computational perspective. *IEEE Intelligent Systems*, 2020.

Dheeru Dua and Casey Graff. UCI machine learning repository. `http://archive.ics.uci.edu/ml`, 2017.

Benjamin Fish, Jeremy Kun, and Ádám D Lelkes. A confidence-based approach for balancing fairness and accuracy. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 144–152. SIAM, 2016.

Naman Goel, Mohammad Yaghini, and Boi Faltings. Non-discriminatory machine learning through convex fairness criteria. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Sara Hajian and Josep Domingo-Ferrer. A methodology for direct and indirect discrimination prevention in data mining. *IEEE transactions on knowledge and data engineering*, 25(7):1445–1459, 2012.

Shunjie Han, Cao Qubo, and Han Meng. Parameter selection in svm with rbf kernel function. In *World Automation Congress 2012*, pages 1–4. IEEE, 2012.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674, 2013.

Lauren Kirchner Jeff Larson, Surya Mattu and Julia Angwin. How we analyzed the compas recidivism algorithm. `https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm`.

László A Jeni, Jeffrey F Cohn, and Fernando De La Torre. Facing imbalanced data–recommendations for the use of performance metrics. In *2013 Humaine association conference on affective computing and intelligent interaction*, pages 245–251. IEEE, 2013.

Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

Niki Kilbertus, Mateo Rojas Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems*, pages 656–666, 2017.

Ming Jun Lai and Abraham Varghese. On convergence of the alternating projection method for matrix completion and sparse recovery problems. *arXiv preprint arXiv:1711.02151*, 2017.

Yoonkyung Lee, Yi Lin, and Grace Wahba. Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data. *Journal of the American Statistical Association*, 99(465):67–81, 2004.

Xuejun Liao, Hui Li, and Lawrence Carin. Generalized alternating projection for weighted-2,1 minimization with applications to model-based compressive sensing. *SIAM Journal on Imaging Sciences*, 7(2):797–823, 2014.

Yi Lin. Support vector machines and the bayes rule in classification. *Data Mining and Knowledge Discovery*, 6(3):259–275, 2002.

Yufeng Liu. Fisher consistency of multicategory support vector machines. In *Artificial intelligence and statistics*, pages 291–298, 2007.

Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency*, pages 107–118, 2018.

Imran Naseem, Roberto Togneri, and Mohammed Bennamoun. Linear regression for face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 32(11):2106–2112, 2010.

Matt Olfat and Anil Aswani. Spectral algorithms for computing fair support vector machines. *arXiv preprint arXiv:1710.05895*, 2017.

Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

Mehdi Pirooznia, Jack Y Yang, Mary Qu Yang, and Youping Deng. A comparative study of different machine learning methods on microarray gene expression data. *BMC genomics*, 9(S1):S13, 2008.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems*, pages 5680–5689, 2017.

Pau Rodriguez, Guillem Cucurull, Jordi Gonzàlez, Josep M Gonfaus, Kamal Nasrollahi, Thomas B Moeslund, and F Xavier Roca. Deep pain: Exploiting long short-term memory networks for facial expression classification. *IEEE transactions on cybernetics*, 2017.

Aihua Shen, Rencheng Tong, and Yaochen Deng. Application of classification models on credit card fraud detection. In *2007 International conference on service systems and service management*, pages 1–4. IEEE, 2007.

Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.

Bahar Taskesen, Viet Anh Nguyen, Daniel Kuhn, and Jose Blanchet. A distributionally robust approach to fair classification. *arXiv preprint arXiv:2007.09530*, 2020.

Sahil Verma and Julia Rubin. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, pages 1–7. IEEE, 2018.

Yongxiong Wang, Xuan Li, and Xueming Ding. Probabilistic framework of visual anomaly detection for unbalanced data. *Neurocomputing*, 201:12–18, 2016.

Petros Xanthopoulos and Talayeh Razzaghi. A weighted support vector machine method for control chart pattern recognition. *Computers & Industrial Engineering*, 70:134–149, 2014.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*, pages 1171–1180, 2017.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P Gummadi. Fairness constraints: A flexible approach for fair classification. *J. Mach. Learn. Res.*, 20 (75):1–42, 2019.

Tong Zhang and Frank J Oles. Text categorization based on regularized linear classification methods. *Information retrieval*, 4(1):5–31, 2001.

Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.