Fair Representation Learning through Implicit Path Alignment

Changjian Shui [1] Qi Chen [1] Jiaqi Li [2] Boyu Wang [2] Christian Gagné [1 3]

Abstract

We consider a fair representation learning perspective, where optimal predictors, on top of the data representation, are ensured to be invariant with respect to different sub-groups. Specifically, we formulate this intuition as a bi-level optimization, where the representation is learned in the outer-loop, and invariant optimal group predictors are updated in the inner-loop. Moreover, the proposed bi-level objective is demonstrated to fulfill the *sufficiency rule*, which is desirable in various practical scenarios but was not commonly studied in the fair learning. Besides, to avoid the high computational and memory cost of differentiating in the inner-loop of bi-level objective, we propose an implicit path alignment algorithm, which only relies on the solution of inner optimization and the implicit differentiation rather than the exact optimization path. We further analyze the error gap of the implicit approach and empirically validate the proposed method in both classification and regression settings. Experimental results show the consistently better trade-off in prediction performance and fairness measurement.

1. Introduction

Machine learning has been widely used in the real world decision-making practice such as job candidate screening (Raghavan et al., 2020). However, it has been observed that learning algorithms treated some groups of population unfavorably, for example, predicting the likelihood of crime on the grounds of ethnicity, gender or age (Hardt et al., 2016). To that end, algorithmic fairness, which aims to mitigate the prediction bias for the *protected feature* such as gender, has recently received tremendous attentions.

With the advancements of deep learning (LeCun et al., 2015), fair representation learning (Zemel et al., 2013) has been recently highlighted. Specifically, the learned fair representation can easily transfer the unbiased prior knowledge to various downstream learning-tasks. For example, in language understanding, the fair embedding provides both useful and unbiased representation for different goals such as translation or recommendation (Chang et al., 2019; Ethayarajh, 2020). Besides, it has been investigated in other scenarios such as computer vision (Kehrenberg et al., 2020) and intelligent health (Fletcher et al., 2021).

Typically, fair representation learning is realized by introducing fair constraints during the training. Consequently, a number of fair notions for various goals have been proposed. Specifically, most existing approaches in classification or regression use *independence* or *separation* rule (see Sec.2 and references therein) (Madras et al., 2018; Song et al., 2019; Chzhen et al., 2020). However, in a variety of applications, independence or separation are not always appropriate, and other fair notions such as *sufficiency rule* (Chouldechova, 2017) are preferred. Intuitively, given the output of the algorithm $\hat{Y}$, the *sufficiency rule* ensures the conditional expectation of label $\mathbb{E}[Y|\hat{Y}]$ is invariant across the different sub-groups (see Sec.2 for the formal definition).

In practice, the negligence of sufficiency rule can lead to the significant bias in intelligent health. For example, health systems rely on commercial algorithms to identify and assist patients with complex health needs. Such algorithms output a score of healthcare needs, where a higher score indicates that the patient is sicker and requires additional care. Notably, Obermeyer et al. (2019) reveals a industry-wide used algorithm that affects millions of patients, exhibits significant racial bias. Under the same predicted score $\hat{Y} = t$, Black patients are considerably sicker than White patients ($\mathbb{E}_{\text{black}}[Y|\hat{Y} = t] > \mathbb{E}_{\text{white}}[Y|\hat{Y} = t]$). Obermeyer et al. (2019) further points out that eliminating this disparity would increase the percentage of Black patients receiving additional healthcare from 17.7% to 46.5%.

From the algorithmic perspective, sufficiency rule is generally non-compatible to independence or separation, as demonstrated in Sec.2, indicating that existing fair algorithms for independence or separation do not improve or even worsen the sufficiency rule. Therefore fair represen-
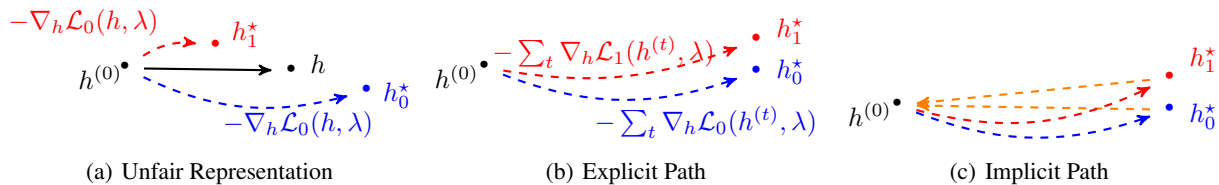
[1]Université Laval, Québec, Canada [2]University of Western Ontario, Ontario, Canada [3]Canada CIFAR AI Chair, Mila. Correspondence to: Boyu Wang <bwang@csd.uwo.ca>, Christian Gagné <christian.gagne@gel.ulaval.ca>.

Figure 1. Illustration of explicit and implicit path. (a) Unfair representation leads to different optimization paths and non-invariant optimal predictors on the latent space $\mathcal{Z}$. (b) The fair representation learning ensures the invariant optimal predictor w.r.t. different sub-groups on $\mathcal{Z}$ (encouraging $h_0^\star = h_1^\star$). Since the gradient based approach is adopted to optimize h, the explicit path alignment aims to learn a representation $\lambda(x)$ to enforce the identical optimization path (i.e, identical blue and red curve) w.r.t. h. (c) The proposed implicit path alignment only requires the last iteration point and approximate the gradient w.r.t. λ from the last update of h (orange arrow).

tation learning w.r.t. the sufficiency rule is important and promising in both practice and algorithmic development.

In this paper, we propose a framework to address the *sufficiency rule* via the following principle: given a fixed representation function, if the *optimal* predictor that learned on the embedding space are *invariant* to different sub-groups, then the corresponding representation function is fair. The principle is further illustrated in Fig. 1(a): when the representation function $\lambda : \mathcal{X} \to \mathcal{Z}$ is unfair and we adopt gradient descent to learn the predictor $h : \mathcal{Z} \to R$. The optimal predictors of different sub-groups (blue, red) are not invariant, yielding biased predictions. Intuitively, the optimal predictor for each subgroup approximates the conditional expectation, which encourages the sufficiency. We will justify such an principle ensures that learned representation could satisfy the sufficiency rule under proper assumptions, showing in Proposition 3.1.

The aforementioned principle can be naturally formulated as a bi-level optimization problem, where we aim to adjust the representation λ (in the outer-loop) to satisfy the invariant optimal predictor h (in the inner-loop). Based on this, when we adopt the gradient-based approach in solving the bi-level objective, a straightforward solution is to learn the representation λ to fulfill the identical *explicit* gradient-descent directions in learning optimal predictor $h^\star$ of different groups, shown in Fig. 1(b). Clearly, if the inner gradient descent step of each sub-group is identical, their final predictors (as the approximation of $h^\star$) will be surely invariant. However, the corresponding algorithmic realization is challenging in deep learning: 1) It requires storing the whole gradient steps, which induces a high memory burden. 2) the embedding function λ is optimized via backpropagation from the whole gradient optimization path, which induces a high computational complexity.

To address this, we propose an implicit path alignment, shown in Fig. 1(c). Namely, we only consider the final (t-th) update of the predictor $h^{(t)}$, then we update representation function λ by approximating its gradient at point $h^{(t)}$ through the implicit function (Bengio, 2000). By using the

gradient approximation, we do not need to store the whole gradient steps and conduct the backpropagation through the entire optimization path. Overall, contributions in this paper are as follows:

Fair representation learning for the sufficiency rule The proposed fair-representation approach is proved to satisfy the sufficiency rule in both classification and regression. We also find such a criteria is intrinsically consistent with the recent proposed Invariant Risk Minimization (Arjovsky et al., 2019; Bühlmann, 2020), which aims to preserve the invariant correlations between the embedding (or representation) and true label. Intuitively, if such correlations are robust and not influenced by the specific sub-group, the learned representation is somehow fair.

Efficient algorithm We propose an implicit path alignment algorithm to learn the fair representation, which address the prohibitive memory and computational cost in the original bi-level objective. We analyze the approximation error gap of the proposed implicit algorithm, which induces a trade-off between the correct gradient estimation and fairness.

Improved fairness in classification and regression We evaluate the implicit algorithm in classification and regression with tabular, computer vision and NLP datasets, where the implicit algorithm effectively improves the fairness.

2. Sufficiency rule

We denote $X \in \mathcal{X}$ as the input, $Y \in \mathcal{Y}$ as the ground truth label, and $\hat{Y} \in \mathcal{Y}$ as algorithm's output. Following the previous work in fair representation learning (Madras et al., 2018), we consider binary protected feature or two sub-groups with corresponding distributions $\mathcal{D}_0$ and $\mathcal{D}_1$. Then according to (Liu et al., 2019), the sufficiency rule is defined as:

$$\mathbb{E}_{\mathcal{D}_0}[Y|\hat{Y} = t] = \mathbb{E}_{\mathcal{D}_1}[Y|\hat{Y} = t], \ \forall t \in \mathcal{Y} \qquad (1)$$

Eq.(1) shows that the conditional expectation of ground truth label Y are identical for $\mathcal{D}_0, \mathcal{D}_1$, given the same prediction output t. Based on Eq.(1), we propose the sufficiency gap

as the metric to measure the fairness. Since we aim to evaluate this in both binary classification ($Y \in \{-1, 1\}$) and regression ($Y \in \mathbb{R}$), the sufficiency gaps are separately defined.

Sufficiency gap in binary classification Based on the sufficiency rule, the sufficiency gap in binary classification is naturally defined as:

$$\Delta\mathrm{Suf}_C = \frac{1}{2} \sum_{y \in \{-1,1\}} |\mathcal{D}_0(Y = y|\hat{Y} = y) - \mathcal{D}_1(Y = y|\hat{Y} = y)|$$

$\Delta\mathrm{Suf}_C \in [0, 1]$ encourages two sub-groups with identical Positive predicted value (PPV) and Negative predicted value (NPV). To better understand this metric, consider the example of healthcare system, which outputs only binary score: *High Risk* or *Low Risk*. Obermeyer et al. (2019) essentially revealed $\mathcal{D}_{\mathrm{black}}(Y = \mathrm{High\ Risk}|\hat{Y} = \mathrm{Low\ Risk}) > \mathcal{D}_{\mathrm{white}}(Y = \mathrm{High\ Risk}|\hat{Y} = \mathrm{Low\ Risk})$: the severity of illness in Black patients is actually underestimated. Thus if $\Delta\mathrm{Suf}_C$ is small, the racial discrimination will be remedied.

Sufficiency gap in regression Based on (Kuleshov et al., 2018), the sufficiency gap in regression is defined as:

$$\Delta\mathrm{Suf}_R = \int_{t \in \mathcal{Y}} |\mathcal{D}_0(Y \leq t|\hat{Y} \leq t) - \mathcal{D}_1(Y \leq t|\hat{Y} \leq t)|dt$$

An illustrative example depicts in Fig. 2. Specifically, $\Delta\mathrm{Suf}_R \in [0, 1]$ is an approximation of $|\mathcal{D}_0(Y = y|\hat{Y} = y) - \mathcal{D}_1(Y = y|\hat{Y} = y)|, \ \forall y \in \mathbb{R}$, since the latter is difficult to estimate since Y is continuous. We also adopt the healthcare example to understand this metric: assuming the health system outputs a real-value healthcare score $\hat{Y} = t$ (higher indicates sicker), Obermeyer et al. (2019); Sjoding et al. (2020) observed $\mathcal{D}_{\mathrm{black}}(Y > t|\hat{Y} \leq t) > \mathcal{D}_{\mathrm{white}}(Y > t|\hat{Y} \leq t)$. Namely, for all the patients with the predicted healthcare score lower than t, the actual sicker proportion ($Y > t$) in Black patients is significantly higher than White patients. Therefore a small $\Delta\mathrm{Suf}_R$ suggests an improved disparity w.r.t. the sufficient rule.

Relation to other fair rules We briefly compare the Sufficiency rule with widely adopted Independence and Separation rule in binary classification. The detailed justifications and comparisons are shown in Appendix.

Independence rule is defined as:

$$\mathbb{E}_{\mathcal{D}_0}[\hat{Y}] = \mathbb{E}_{\mathcal{D}_1}[\hat{Y}],$$

In binary classification, the Independence rule is also referred as *demographic parity* (DP) (Zemel et al., 2013). We can further justify that if $\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y)$ (i.e, different label distribution in the sub-groups), the Sufficiency and Independence rule cannot both hold.
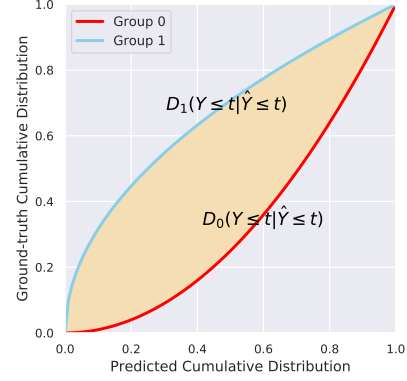

Figure 2. Illustrative example of Sufficiency gap ($\Delta\mathrm{Suf}_R$) in regression

Separation Rule is defined as:

$$\mathbb{E}_{\mathcal{D}_0}[\hat{Y}|Y = t] = \mathbb{E}_{\mathcal{D}_1}[\hat{Y}|Y = t], \ \ \forall t \in \mathcal{Y}$$

In binary classification, the Separation rule is also denoted as *Equalized Odds* (EO) (Hardt et al., 2016). Barocas et al. (2019) further justified that if $\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y)$ and the joint distribution of $(Y, \hat{Y})$ has positive probability in $\mathcal{D}_0, \mathcal{D}_1$, the Sufficiency and Separation rule cannot both hold.

3. Fair representation learning as a bi-level optimization

We denote the representation function λ that maps the input X into the latent variable $Z \in \mathcal{Z}$, the prediction function h such that $h : \mathcal{Z} \to \mathbb{R}$ for regression and $h : \mathcal{Z} \to \{-1, 1\}$ for binary classification. We denote the prediction loss as ℓ, the prediction loss on sub-group $\mathcal{D}_0, \mathcal{D}_1$ is expressed as:

$$\mathcal{L}_0(h, \lambda) = \mathbb{E}_{(x,y)\sim\mathcal{D}_0} \ell(h \circ \lambda(x), y)$$
$$\mathcal{L}_1(h, \lambda) = \mathbb{E}_{(x,y)\sim\mathcal{D}_1} \ell(h \circ \lambda(x), y)$$

According to the intuition, we aim to solve the following bi-level objective:

$$\min_{\lambda} \ \mathcal{L}_0(h_0^\star, \lambda) + \mathcal{L}_1(h_1^\star, \lambda) \qquad \text{(Outer-Loop)}$$
$$\text{s.t. } h_0^\star = h_1^\star, \qquad\qquad\qquad \text{(Inner-Loop)}$$
$$h_0^\star \in \underset{h}{\mathrm{argmin}}\ \mathcal{L}_0(h, \lambda), \ h_1^\star \in \underset{h}{\mathrm{argmin}}\ \mathcal{L}_1(h, \lambda).$$

In the outer-loop, we aim to find a representation function λ for minimizing the prediction error, given the optimal predictor $(h_0^\star, h_1^\star)$ on the embedding space $\mathcal{Z}$. As for the inner-loop, given a fixed representation λ, $h_0^\star, h_1^\star$ are the optimal predictor for each sub-group. The constraints $h_0^\star = h_1^\star$ additionally encourage the invariant optimal predictors from $\mathcal{D}_0, \mathcal{D}_1$.

Relation to explicit path alignment In deep learning, we adopt gradient-based approaches to minimize the loss, therefore $h^\star$ in the inner-loop is approximated as $h^{(t+1)}$, the t-th update in the gradient descent: $h_0^\star \approx h^{(0)} - \sum_t \nabla_h \mathcal{L}_0(h^{(t)}, \lambda), h_1^\star \approx h^{(0)} - \sum_t \nabla_h \mathcal{L}_1(h^{(t)}, \lambda)$, where $h^{(0)}$ is the common initialization. Thus the invariant optimal predictor is equivalent to:

$$\sum_t \nabla_h \mathcal{L}_0(h^{(t)}, \lambda) = \sum_t \nabla_h \mathcal{L}_1(h^{(t)}, \lambda).$$

The aforementioned equation suggests learning a representation λ that ensures the identical optimization path w.r.t. h for each sub-group, which recovers the explicit path alignment.

Relation to Sufficiency rule We further demonstrate the relation between the bi-level objective and Sufficiency rule.

Proposition 3.1. *If we specify the prediction loss ℓ as logistic regression loss in the classification $\log(1 + \exp(-yh(z)))$ with $\mathcal{Y} = \{-1, 1\}$ and the square loss in the regression $(h(z) - y)^2$ with $\mathcal{Y} \subset \mathbb{R}$. Then minimizing the inner-loop loss is equivalent to:*

$$\mathbb{E}_{\mathcal{D}_0}[Y|Z = z] = \mathbb{E}_{\mathcal{D}_1}[Y|Z = z],$$
$$\mathbb{E}_{\mathcal{D}_0}[Y|\hat{Y} = h^\star(z)] = \mathbb{E}_{\mathcal{D}_1}[Y|\hat{Y} = h^\star(z)]$$

where $h^\star = h_0^\star = h_1^\star$ and $z = \lambda(x)$.

Proposition 3.1 demonstrates that the objective of inner-loop loss fulfills the sufficiency rule in both binary classification and regression.

4. Proposed Algorithms

We propose the implicit alignment in deep learning, where λ and h are implemented by the neural network. We also reformulate as the original objective through *Lagrangian relaxation*:

$$\min_\lambda \; \mathcal{L}_0(h_0^\star, \lambda) + \mathcal{L}_1(h_1^\star, \lambda) + \frac{\kappa}{2}\|h_0^\star - h_1^\star\|_2^2$$
(Outer-Loop)
$$\text{s.t. } h_0^\star \in \underset{h}{\arg\min}\, \mathcal{L}_0(h, \lambda), \; h_1^\star \in \underset{h}{\arg\min}\, \mathcal{L}_1(h, \lambda),$$
(Inner-Loop)

where the introduced $\kappa > 0$ is the coefficient to control the fairness, with a sufficient large κ ensuring $h_0^\star \approx h_1^\star$. Then we drive the approximated gradient w.r.t. λ, which contains the following key elements.

Solving the inner optimization Given a fixed representation λ, we find $h_0^\epsilon, h_1^\epsilon$ such that:

$$\|h_0^\star - h_0^\epsilon\| \leq \epsilon, \quad \|h_1^\star - h_1^\epsilon\| \leq \epsilon,$$

where ϵ is the optimization tolerance. Besides, $h_1^\star$ and h_1^ϵ are essentially the function of λ, i.e., h_1^ϵ depends on the

predefined representation function λ. It is worth mentioning that the optimization tolerance ϵ is realistic. E.g, consider a fixed representation and one-layer predictor h_0, h_1, the optimization will be convex.

Computing the gradient of λ Given the approximate solution $h_0^\epsilon, h_1^\epsilon$, we can compute the gradient w.r.t. λ (referred as $\tilde{\text{grad}}(\lambda)$) [1] in the outer-loop:

$$\begin{aligned}\tilde{\text{grad}}(\lambda) =& \nabla_\lambda \mathcal{L}_0(h_0^\epsilon, \lambda) + \nabla_\lambda \mathcal{L}_1(h_1^\epsilon, \lambda) \\ & (\nabla_\lambda h_0^\epsilon)^T \left(\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda) + \kappa(h_0^\epsilon - h_1^\epsilon)\right) \\ & + (\nabla_\lambda h_1^\epsilon)^T \left(\nabla_{h_1}\mathcal{L}_1(h_1^\epsilon, \lambda) - \kappa(h_0^\epsilon - h_1^\epsilon)\right).\end{aligned}$$

Where $\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda)$ is the partial derivative in the loss w.r.t. the first term (about h_0), evaluated at h_0^ϵ. Also $\nabla_\lambda \mathcal{L}_0(h_0^\epsilon, \lambda)$ is the partial derivative w.r.t. the second term (about λ).

Implicit function for approximating the gradient In order to compute $\tilde{\text{grad}}(\lambda)$ in `autograd`, we need to estimate $\nabla_\lambda h_0^\epsilon$ and $\nabla_\lambda h_1^\epsilon$. We herein adopt the implicit function (Bengio, 2000) to approximate $\nabla_\lambda h_0^\epsilon$, which has been adopted in the hyperparameter optimization (Pedregosa, 2016) and meta-learning (Rajeswaran et al., 2019).

Concretely, if the prediction loss is smooth and there exist stationary points to achieve optimal, we have: $\nabla_{h_0}\mathcal{L}_0(h_0^\star(\lambda), \lambda) = 0, \nabla_{h_1}\mathcal{L}_0(h_1^\star(\lambda), \lambda) = 0$. Then differentiating w.r.t. λ will induce: $\mathbf{d}\left(\nabla_{h_0}\mathcal{L}_0(h_0^\star(\lambda), \lambda)\right)/\mathbf{d}\lambda = \nabla_{h_0}^2\mathcal{L}_0(h_0^\star, \lambda)\nabla_\lambda h_0^\star + \nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^\star, \lambda) = 0$.[2] Thus we have $\nabla_\lambda h_0^\star = -\left(\nabla_{h_0}^2\mathcal{L}_0(h_0^*, \lambda)\right)^{-1}\left(\nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^*, \lambda)\right)$, where the Hessian matrix $\nabla_{h_0}^2\mathcal{L}_0(h_0^*, \lambda)$ is assumed to be invertible.

Through the implicit function, we can approximate $\nabla_\lambda h_0^\epsilon$ as:

$$\nabla_\lambda h_0^\epsilon \approx -\left(\nabla_{h_0}^2\mathcal{L}_0(h_0^\epsilon, \lambda)\right)^{-1}\left(\nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda)\right)$$

As for $\nabla_\lambda h_1^\epsilon$, we have the similar result: $\nabla_\lambda h_1^\epsilon \approx -\left(\nabla_{h_1}^2\mathcal{L}_1(h_1^\epsilon, \lambda)\right)^{-1}\left(\nabla_\lambda\nabla_{h_1}\mathcal{L}_1(h_1^\epsilon, \lambda)\right)$.

Efficient and numerical stable gradient estimation Plugging in the approximations, the gradient w.r.t λ is approximated as:

$$\tilde{\text{grad}}(\lambda) \approx \nabla_\lambda\mathcal{L}_0(h_0^\epsilon, \lambda) - \left(\nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda)\right)^T\mathbf{p}_0 + \nabla_\lambda\mathcal{L}_1(h_1^\epsilon, \lambda) - \left(\nabla_\lambda\nabla_{h_1}\mathcal{L}_1(h_1^\epsilon, \lambda)\right)^T\mathbf{p}_1$$

Where $\mathbf{p}_0, \mathbf{p}_1$ are denoted as the inverse-Hessian vector product with:

$$\mathbf{p}_0 = \left(\nabla_{h_0}^2\mathcal{L}_0(h_0^\epsilon, \lambda)\right)^{-1}\left(\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda) + \kappa(h_0^\epsilon - h_1^\epsilon)\right)$$
$$\mathbf{p}_1 = \left(\nabla_{h_1}^2\mathcal{L}_1(h_1^\epsilon, \lambda)\right)^{-1}\left(\nabla_{h_1}\mathcal{L}_1(h_1^\epsilon, \lambda) - \kappa(h_0^\epsilon - h_1^\epsilon)\right).$$

---

[1]We denote the ground truth gradient as $\text{grad}(\lambda)$ if we adopt optimal predictor $h_0^\star, h_1^\star$ in the computation.

[2]$\mathbf{d}(\cdot)/\mathbf{d}\lambda$ denotes the total derivative.

Algorithm 1 Implicit Path Alignment Algorithm

1: **Input:** Representation function λ, predictor h_0, h_1, datasets from two sub-groups $\mathcal{D}_0$. $\mathcal{D}_1$.
2: **for** mini-batch of samples from $(\mathcal{D}_0, \mathcal{D}_1)$ **do**
3:    Solving the inner-loop optimization with tolerance ϵ. Obtaining $h_0^\epsilon, h_1^\epsilon$.
4:    Solving Eq. (2), (3) with tolerance δ. Obtaining $\mathbf{p}_0^\delta$ and $\mathbf{p}_1^\delta$.
5:    Computing $\tilde{\mathrm{grad}}^\delta(\lambda)$ (gradient of representation λ)
6:    Updating λ through `autograd`: $\lambda \leftarrow \lambda - \tilde{\mathrm{grad}}^\delta(\lambda)$
7: **end for**
8: **Return:** $\lambda, h_0^\epsilon, h_1^\epsilon$

However, the current form is still computationally expensive due to the computation of inverse Hessian matrix. Then computing $\mathbf{p}_0$ and $\mathbf{p}_1$ is equivalent to solve the following quadratic programming (QP):

$$\mathrm{argmin}_{\hat{\mathbf{p}}_0} \quad \frac{1}{2}\hat{\mathbf{p}}_0^T \left(\nabla_{h_0}^2 \mathcal{L}_0(h_0^\epsilon, \lambda)\right) \hat{\mathbf{p}}_0$$
$$- \hat{\mathbf{p}}_0^T \left(\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda) + \kappa(h_0^\epsilon - h_1^\epsilon)\right) \quad (2)$$

$$\mathrm{argmin}_{\hat{\mathbf{p}}_1} \quad \frac{1}{2}\hat{\mathbf{p}}_1^T \left(\nabla_{h_1}^2 \mathcal{L}_1(h_1^\epsilon, \lambda)\right) \hat{\mathbf{p}}_1$$
$$- \hat{\mathbf{p}}_1^T \left(\nabla_{h_1}\mathcal{L}_1(h_1^\epsilon, \lambda) - \kappa(h_0^\epsilon - h_1^\epsilon)\right) \quad (3)$$

Since it is a typical QP problem and we adopt conjugate gradient method (Concus et al., 1985; Rajeswaran et al., 2019), which can be updated efficiently through `autograd` via computing the Hessian-vector product. We additionally suppose the optimization error in the QP as δ, i.e.: $\|\mathbf{p}_0 - \mathbf{p}_0^\delta\| \le \delta$, $\|\mathbf{p}_1 - \mathbf{p}_1^\delta\| \le \delta$, then the gradient w.r.t representation λ can be finally expressed as:

$$\tilde{\mathrm{grad}}^\delta(\lambda) = \nabla_\lambda\mathcal{L}_0(h_0^\epsilon, \lambda) - \left(\nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda)\right)^T \mathbf{p}_0^\delta + \nabla_\lambda\mathcal{L}_1(h_1^\epsilon, \lambda) - \left(\nabla_\lambda\nabla_{h_1}\mathcal{L}_1(h_1^\epsilon, \lambda)\right)^T \mathbf{p}_1^\delta$$

The $\tilde{\mathrm{grad}}^\delta(\lambda)$ can be also efficiently estimated through Hessian vector product via `autograd` without explicitly computing the Hessian matrix.

Proposed algorithm   Based on the key elements, the proposed algorithm is shown in Algo. 1.

4.1. The cost of Implicit algorithm: Approximation-Fair Trade-off

In the proposed objective bi-level loss, a sufficient large κ encourages the invariant optimal predictor, yielding the fair results. However, the implicit approach will lead to a biased estimation of the ground truth gradient. We analyze the error gap of the approximation in Theorem 4.1.

Theorem 4.1 (Approximation Error Gap). *Suppose that (1) Smooth Predictive Loss. The first-order derivatives and second-order derivatives of $\mathcal{L}$ are Lipschitz continuous; (2) Non-singular Hessian matrix. We assume $\nabla_{h_0, h_0}\mathcal{L}_0(h_0, \lambda), \nabla_{h_1, h_1}\mathcal{L}_1(h_1, \lambda)$, the Hessian matrix of the inner optimization problem, are invertible. (3) Bounded representation and predictor function. We assume the λ and h are bounded, i.e., $\|\lambda\|, \|h\|$ are upper bounded by the predefined positive constants. Then the approximation error between the ground truth and algorithmic estimated gradient w.r.t. the representation is be upper bounded by:*

$$\|grad(\lambda) - \tilde{grad}^\delta(\lambda)\| = \mathcal{O}(\kappa\epsilon + \epsilon + \delta).$$

The proof is delegated in Appendix C. We also discuss the assumptions to guarantee the convergence of Algorithm 1, shown in Appendix D.

Theorem 4.1 reveals that the gradient approximation error depends on the two-level optimization tolerance ϵ, δ and the coefficient of fair constraints κ. Specifically, the error gap reveals the inherent trade-off in accurate gradient estimation and fair-representation learning. If we fix the optimization tolerance ϵ and δ, a smaller κ indicates a better approximation of the gradient, which yields weak fair constraints. Thus the implicit alignment introduces a trade-off in the prediction performance (i.e., correct approximation of the gradient) and fairness measurement.

5. Related Work

Fair Machine Learning   Below we only list the most related work and refer to the survey paper (Mehrabi et al., 2021) for details in the algorithmic fairness. In the *classification*, various methods in learning fair representations have been proposed. Specifically, a common strategy is to introduce the statistical constraints as the regularization during the training, e.g., demographic parity (DP) (Zhang et al., 2018; Madras et al., 2018; Song et al., 2019; Jiang et al., 2020; Kehrenberg et al., 2020) that encourages the identical output of the representation or equalized odds (EO) (Song et al., 2019; Gupta et al., 2021) that ensures the identical conditional output of the representation, given the ground truth label Y. Another direction is to disentangle the data for factorizing meaningful representations such as (Locatello et al., 2019; Kim et al., 2019). Intuitively, the disentangled embedding is independent of the protected feature, thus reflecting a fair representation w.r.t. the independence rule, which can be potentially problematic when the label distributions of sub-groups vary dramatically (Zhao et al., 2019).

The concept of fairness has also been extended to the fields beyond classification. For instance, in the *regression* problem (Komiyama et al., 2018; Agarwal et al., 2019), the bounded group loss has been proposed as the fair measure:

if the prediction loss in each sub-group is smaller than ϵ, the regression is ϵ-level fair. In fact, the fair criteria in our paper is *not equivalent* to ϵ-fair. Considering a fixed representation function λ, the ϵ-level fair does not guarantee the *optimal* and *invariant* predictor for each sub-group and vice versa.

The sufficiency rule has also been discussed in the previous work. Notably, Chouldechova (2017); Liu et al. (2019) proposed the sufficiency gap *in classification* for measuring fairness w.r.t. the sufficiency rule. (Liu et al., 2019) also discussed the relations between the sufficiency gap and probabilistic calibration (Guo et al., 2017) (referred as calibration gap). According to Pleiss et al. (2017), the calibration rule is a stronger condition than sufficiency rule while it can simultaneously hurt the prediction performance. Throughout this paper, we only consider the sufficiency rule. The triple trade-off between the probabilistic calibration, sufficiency rule and accuracy will be left as future work.

Learning Invariance The analyzed fair-representation criteria shares a quite similar spirit to the IRM or Invariant Risk Minimization (Arjovsky et al., 2019; Bühlmann, 2020; Creager et al., 2021), where an algorithm IRM_v1 is proposed to enable the out-of-distribution (OOD) generalization. The key difference between our work and (Arjovsky et al., 2019) lies in the algorithmic aspect: it has been theoretically justified that the originally proposed IRM_v1 does not necessarily capture the invariance across the environments (Rosenfeld et al., 2020; Shui et al., 2022). By contrast, we aim to solve the bi-level objective in the context of deep-learning and propose an efficient and principled practical algorithm with better empirical performance than IRM_v1. Besides, based on results of (Chen et al., 2021), the proposed algorithm does not provably guarantee the OOD generalization property due to the limited sub-groups ($N = 2$) considered within the paper.

6. Experiments

6.1. Experimental setup

In the paper, we adopt the aforementioned sufficiency gap as fair metrics, where $\hat{Y}$ is denoted as:

$$\hat{Y} = \begin{cases} h_0^\epsilon \circ \lambda(X), & X \in \mathcal{D}_0 \\ h_1^\epsilon \circ \lambda(X), & X \in \mathcal{D}_1 \end{cases}$$

Then in the binary classification, we can estimate $\Delta\mathrm{Suf}_C = \frac{1}{2}\sum_{y\in\{-1,+1\}}|\mathcal{D}_0(Y = y|\hat{Y} = y) - \mathcal{D}_1(Y = y|\hat{Y} = y)|$ from the data.

As for regression, the original form $\Delta\mathrm{Suf}_R = \int_t |\mathcal{D}_0(Y \leq t|\hat{Y} \leq t) - \mathcal{D}_1(Y \leq t|\hat{Y} \leq t)|$ (as shown in Fig. 2, Appendix) is difficult to estimate due to the integration term. To address this, we sample multiple values $\{t_1, \ldots, t_m\}$

and compute its average differences as the approximation of the integration. Namely, $\Delta\mathrm{Suf}_R \approx \frac{1}{m}\sum_{i=1}^m |\mathcal{D}_0(Y \leq t_i|\hat{Y} \leq t_i) - \mathcal{D}_1(Y \leq t_i|\hat{Y} \leq t_i)|$. Concretely, for a given t_i in each group, we compute the percentile ($\hat{Y}_0$) at point t: $\mathcal{D}_0(\hat{Y}_0 \leq t_i)$, then we compute the corresponding ground truth cumulative distribution (Y) at the same point t_i: $\mathcal{D}(Y \leq t_i|\hat{Y} \leq t_i)$. Through the aforementioned approximation, we can estimate $|\mathcal{D}_0(Y \leq t_i|\hat{Y} \leq t_i) - \mathcal{D}_1(Y \leq t_i|\hat{Y} \leq t_i)|$.

Baselines We consider the baselines that add fairness constraints during the training process. Specifically, we compare our method with (I) Empirical Risk Minimization (ERM) that trains the model without considering fairness; (II) Adversarial Debiasing (referred as adv_debias) (Zhang et al., 2018); (III) Fair Mix-up (Chuang & Mroueh, 2021), a recent data-augmentation and effective approach in the fair representation learning. In fact, the baselines (II) and (III) are based on Independence rule or Demographic-Parity (DP), which is designed to demonstrate the general non-compatibility in addressing the sufficiency rule.

Besides, we include two additional baselines that have the similar objective but different algorithmic realizations. (IV) the original IRM regularization (referred as IRM_v1) (Arjovsky et al., 2019), which adds a gradient penalty to encourage the invariance among the different groups. (V) One-step explicit alignment. In the inner-loop optimization, we suppose to conduct a simple one-step gradient descent ($T = 1$) for each sub-group, i.e, $h_0^\star \approx h_{\mathrm{init}} - \nabla_{h_0}\mathcal{L}_0(h_0, \lambda)$, $h_1^\star \approx h_{\mathrm{init}} - \nabla_{h_1}\mathcal{L}_1(h_1, \lambda)$. Thus in the outer-loop optimization, we add a gradient-incoherence constraint to encourage the identical (one-step) optimization path: $\min_\lambda \|\nabla_{h_0}\mathcal{L}_0(h_0, \lambda) - \nabla_{h_1}\mathcal{L}_1(h_1, \lambda)\|_2^2$.

All the results are reported by averaging five repetitions and additional experimental details are delegated in the Appendix.

6.2. Toxic Comments

The toxic comments dataset (Jigsaw, 2018) is a binary **classification** task in NLP to predict whether comment is toxic or not. The original label is actually not binary since the comments is decided by multiple annotators, where the labelling discrepancy generally occurs. To this end, we conduct a simple strategy to decide comment is toxic if at least one annotator marks it. In this dataset, a portion of comments have been labeled with identity attributes, including gender and race. It has also been revealed that the race identity (e.g., black) is correlated with the toxicity label, which can lead to the predictive discrimination. Thus we adopted the *race* as the protected feature by selecting two sub-groups of Black and Asian. For the sake of computational simplicity, we first applied the pretrained BERT (Devlin et al., 2018) to extract

Table 1. Fair Classification. Accuracy and $\Delta\mathrm{Suf}_C$ in Toxic comments (left) and CelebA datasets (right)

Toxic comments	Accuracy ($\uparrow$)	$\Delta\mathrm{Suf}_C$ ($\downarrow$)	CelebA	Accuracy ($\uparrow$)	$\Delta\mathrm{Suf}_C$ ($\downarrow$)
ERM (I)	0.768 ± 0.004	0.173 ± 0.008	ERM (I)	0.780 ± 0.015	0.210 ± 0.022
Adv_debias (II)	0.760 ± 0.008	0.291 ± 0.006	Adv_debias (II)	0.785 ± 0.022	0.165 ± 0.028
Mixup (III)	0.758 ± 0.003	0.343 ± 0.022	Mixup (III)	0.792 ± 0.011	0.160 ± 0.010
IRM_v1 (IV)	0.753 ± 0.004	0.057 ± 0.015	IRM_v1 (IV)	0.795 ± 0.012	0.086 ± 0.015
One_step (V)	0.755 ± 0.007	0.048 ± 0.008	One_step (V)	0.797 ± 0.006	0.086 ± 0.012
Implicit	0.760 ± 0.007	0.051 ± 0.012	Implicit	0.794 ± 0.027	0.074 ± 0.020

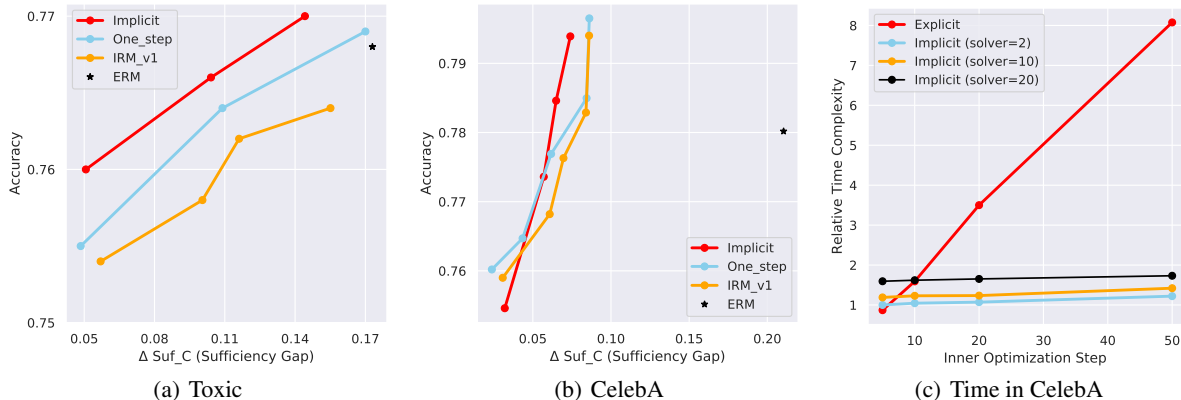(a) Toxic   (b) CelebA   (c) Time in CelebA

Figure 3. Fair Classification. (a,b) The Accuracy-Fair trade-off curve in Toxic (a) and CelebA (b) dataset. The implicit approach demonstrated a consistently better trade-off. (c) Running time comparison of Explicit and Implicit alignment in CelebA dataset. Specifically, solver = 2 indicates that the conjugate-gradient algorithm is executed 2 iterations. The results show that implicit approach avoids the long back-propagation of the entire inner-optimization path. The time complexity of the explicit approach, on the other hand, increases linearly with the inner-optimization step.

the word embedding with 748 dimensional vector. Then we adopt representation function λ as two fully-connected layers with hidden dimension 200 with Relu activation and classifier h as a linear predictor. We report the test-set sub-group average accuracy and sufficiency gap ($\Delta\mathrm{Suf}_C$) in Tab. 1 and Fig. 3(a).

From the results, the Demographic Parity (DP) based fair constraints are non-compatible with the sufficiency rule. Specifically, baseline (II,III) even increase $\Delta\mathrm{Suf}_C$ with higher value than ERM. For the baselines that track the sufficiency rule (IV,V), the sufficiency gap $\Delta\mathrm{Suf}_C$ is improved with a similar accuracy, shown in Tab.1. We also change the regularization coefficient in (IV,V) and κ in the implicit approach. We observe that the implicit approach demonstrates a consistent better Accuracy-Fair trade-off, shown in Fig. 3(a).

6.3. CelebA Dataset

The CelebA dataset (Liu et al., 2015) contains around 200K images of celebrity faces, where each image is associated with 40 human-annotated binary attributes including gender, hair color, young, etc. In this paper, we designate *gender* as

the protected feature, and *attractive* as the binary **classification** task. We randomly select around 82K and 18K images as the training and validation set. Then we adopt representation function λ as pre-trained ResNet-18 (He et al., 2016) and classifier h as two-fully connected layers. We report the test-set sub-group average accuracy and sufficiency gap ($\Delta\mathrm{Suf}_C$) in Tab. 1 and Fig. 3(b).

The results in the CelebA show similar behaviors with the Toxic comments. Specifically, the DP based fair approaches (II, III) did not effectively improve $\Delta\mathrm{Suf}_C$, shown in Tab. 1. In contrast, the sufficiency can be significantly improved in baselines (IV, V) and implicit approach without largely losing the accuracy. Specifically, Fig. 3(b) visualizes the accuracy-fair trade-off curve, where the later three approaches show quite similar behaviors.

Ablation: Computational benefits of Implicit Alignment
To show the efficiency of implicit approach in deep neural network, we empirically evaluated the computing time of T-inner step explicit alignment and implicit approach. The experimental results (shown in Fig.3(c)) verified the computational efficiency of Implicit alignment. Notably, a large

Table 2. Fair Regression. MSE and $\Delta\mathrm{Suf}_R$ in Law dataset (left) and NLSY dataset (right)

Law	MSE ($\downarrow$)	$\Delta\mathrm{Suf}_R$ ($\downarrow$)	NLSY	MSE ($\downarrow$)	$\Delta\mathrm{Suf}_R$ ($\downarrow$)
ERM (I)	0.190 ± 0.005	0.160 ± 0.007	ERM (I)	1.939 ± 0.021	0.246 ± 0.019
Adv_debias (II)	0.223 ± 0.008	0.188 ± 0.012	Adv_debias (II)	1.982 ± 0.016	0.252 ± 0.020
Mixup (III)	0.216 ± 0.012	0.172 ± 0.007	Mixup (III)	1.979 ± 0.025	0.246 ± 0.023
IRM_v1 (IV)	0.208 ± 0.006	0.096 ± 0.006	IRM_v1 (IV)	1.927 ± 0.031	0.077 ± 0.009
One_step (V)	0.204 ± 0.007	0.125 ± 0.010	One_step (V)	1.904 ± 0.027	0.090 ± 0.019
Implicit	0.198 ± 0.005	0.091 ± 0.011	Implicit	1.906 ± 0.019	0.051 ± 0.005

(a) ERM (b) Fair Mix-up (c) Implicit
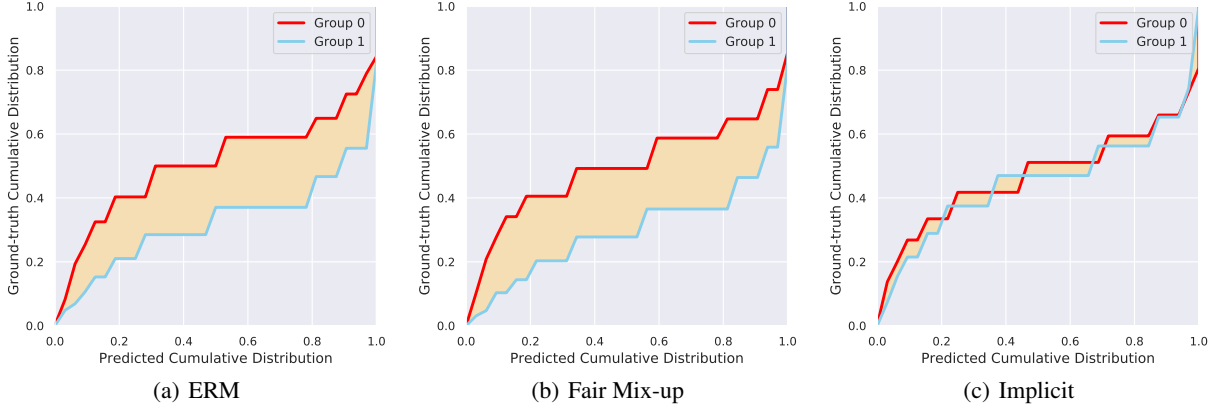
Figure 4. Illustration of the sufficiency gap ($\Delta\mathrm{Suf}_R$) in Law dataset (regression). The ERM and Fair mix-up suffer a high $\Delta\mathrm{Suf}_R$, while the proposed implicit alignment can significantly mitigate the sufficiency gap.

inner-optimization step does not considerably increase the whole computational time of implicit approach with different iterations of conjugated gradient solver. In contrast, the corresponding computational time complexity in explicit alignment linearly scales with the inner-optimization steps, which is consistent with our analysis.

6.4. Law Dataset

The Law Dataset is a **regression** task to predict a students GPA (real value, ranging from $[0, 4]$), where the data is utilized from the School Admissions Councils National Longitudinal Bar Passage Study (Wightman, 1998) with 20K examples. In the regression task, we adopt the square loss and *race* as the protected feature (white versus non-white). We adopt λ as the one fully connected layer with hidden dimension 100 and Relu activation and predictor h as a linear predictor. We report the test-set sub-group average MSE (Mean Square Error) and sufficiency gap ($\Delta\mathrm{Suf}_R$) in Tab. 2 and Fig. 5.

Compared to the classification task, the results show similar behaviors in the regression. Specifically, the DP based fair approaches (II, III) still increase $\Delta\mathrm{Suf}_R$ in the regression. In contrast, the gap is significantly improved in our proposed approach and baseline (IV,V). Specifically, Fig. 4 visualizes
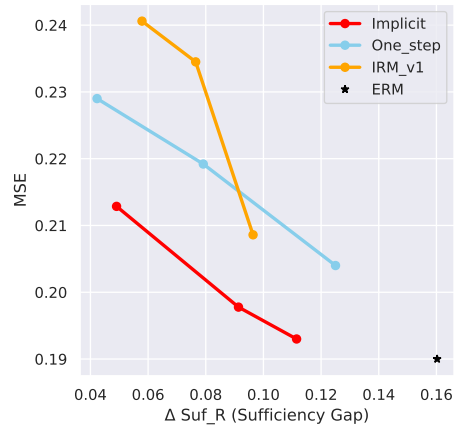

Figure 5. Law Dataset (regression). MSE-Fair Trade-off curve

the sufficiency-gap of different approaches, where the implicit approach significantly mitigates the sufficiency gap. In addition, Fig. 5 describes the MSE-sufficiency gap curve, which further justifies the benefits of implicit approach with a better trade-off between the prediction performance and fairness.

6.5. NLSY Dataset

The National Longitudinal Survey of Youth (NLSY, 2021) dataset is a **regression** task with around 7K dataset, which involves the survey results of the U.S. Bureau of Labor Statistics. It is intended to gather information on the labor market activities and other life events of several groups for predicting the income y of each person. We treat the *gender* as the protected feature. We also normalize the output y by diving the $10,000$, then the final output y ranges around $[0, 8]$. The prediction loss is also the square loss. We adopt representation λ as the two fully connected layers with hidden dimension 200 and Relu activation and predictor h as a linear predictor. We report the testset sub-group average MSE (Mean Square Error) and Sufficiency Gap (ΔSuf_R) in Tab. 2 and Fig. 6.

Tab. 2 provides similar trends with other datasets. Baselines (IV,V) and implicit approach effective control the sufficiency gap, while the DP based approach generally fails to improve the gap. Fig. 6 reveals a slightly better approximation-fair trade off for the implicit approach. Finally, Fig. 8 (in Appendix) visualizes the sufficiency gap of different algorithms. The gap is actually significantly improved while the calibration gap still exists, which is consistent with (Liu et al., 2019). Therefore it can be quite interesting and promising to analyze the triple trade-off between the sufficiency gap, probabilistic calibration and prediction performance in the regression in the future.
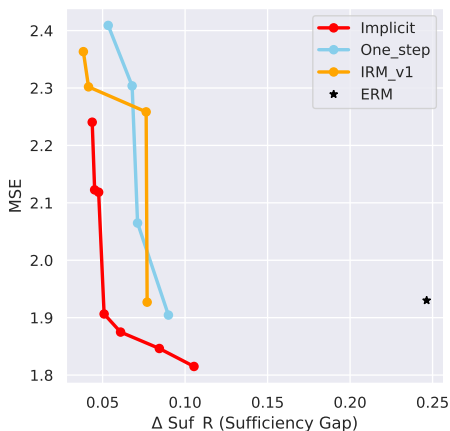

Figure 6. NLSY (regression). MSE-Fair Trade-off curve

7. Conclusion

We considered the fair representation learning from a novel perspective through encouraging the invariant optimal predictors on the top of data representation. We formulated this problem as a bi-level optimization and proposed an implicit alignment algorithm. We further demonstrated the bi-level objective is to fulfil the Sufficiency rule. Then we analyzed

the error gap of the implicit algorithm, which reveal the trade-off of biased gradient approximation and fairness constraints. The empirical results in both classification and regression settings suggest the consistently improved fairness measurement. Finally, we think the future work can include developing computationally efficient explicit algorithms for avoiding the biased gradient computation.

Limitations

We considered a novel fair representation learning perspective to encourage the sufficiency rule. Simultaneously this work remains several limitations.

In the proposed algorithm, we need a two-step optimization with tolerance ϵ and δ. As for controlling ϵ (the tolerance w.r.t. the predictor h), since h is a shallow network with one or two layers, then optimizing over h will be relatively easy. As for δ, since the representation λ could be highly non-convex and high-dimensional, controlling δ would be quite difficult in theory. In practice, we generally control the steps in the conjugate gradient while it is unclear the convergence behavior in the highly non-convex settings.

The current paper mainly focus the binary sensitive attribute with two subgroups. Although it is feasible to extend the multi-attribute settings by consider the pair-wise path alignment, but it would be promising to consider an efficient algorithm for the multi-attribute.

The performance and fair trade-off is induced by the efficient gradient estimation in the bi-level objective. Thus it would be promising to develop an efficient explicit path approach for avoiding such a trade-off.

Acknowledgments

References

Agarwal, A., Dudík, M., and Wu, Z. S. Fair regression: Quantitative definitions and reduction-based algorithms. In *International Conference on Machine Learning*, pp. 120–129. PMLR, 2019.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Barocas, S., Hardt, M., and Narayanan, A. *Fairness and Machine Learning*. fairmlbook.org, 2019. `http://www.fairmlbook.org`.

Bengio, Y. Gradient-based optimization of hyperparameters. *Neural computation*, 12(8):1889–1900, 2000.

Boyd, S., Boyd, S. P., and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Bühlmann, P. Invariance, causality and robustness. *Statistical Science*, 35(3):404–426, 2020.

Chang, K.-W., Prabhakaran, V., and Ordonez, V. Bias and fairness in natural language processing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): Tutorial Abstracts*, Hong Kong, China, November 2019. Association for Computational Linguistics. URL `https://aclanthology.org/D19-2004`.

Chen, Y., Rosenfeld, E., Sellke, M., Ma, T., and Risteski, A. Iterative feature matching: Toward provable domain generalization with logarithmic environments. *arXiv preprint arXiv:2106.09913*, 2021.

Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

Chuang, C.-Y. and Mroueh, Y. Fair mixup: Fairness via interpolation. *arXiv preprint arXiv:2103.06503*, 2021.

Chzhen, E., Denis, C., Hebiri, M., Oneto, L., and Pontil, M. Fair regression with wasserstein barycenters. *arXiv preprint arXiv:2006.07286*, 2020.

Concus, P., Golub, G., and Meurant, G. Block preconditioning for the conjugate gradient method. *Siam Journal on Scientific and Statistical Computing*, 6, 01 1985. doi: 10.1137/0906018.

Creager, E., Jacobsen, J.-H., and Zemel, R. Environment inference for invariant learning. In *International Conference on Machine Learning*, pp. 2189–2200. PMLR, 2021.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Ethayarajh, K. Is your classifier actually biased? measuring fairness under uncertainty with bernstein bounds. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2914–2919, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.262. URL `https://aclanthology.org/2020.acl-main.262`.

Fletcher, R. R., Nakeshimana, A., and Olubeko, O. Addressing fairness, bias, and appropriate use of artificial intelligence and machine learning in global health. *Frontiers in Artificial Intelligence*, 3:116, 2021. ISSN 2624-8212. doi: 10.3389/frai.2020.561802. URL `https://www.frontiersin.org/article/10.3389/frai.2020.561802`.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017.

Gupta, U., Ferber, A., Dilkina, B., and Ver Steeg, G. Controllable guarantees for fair outcomes via contrastive information estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7610–7619, 2021.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29:3315–3323, 2016.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jiang, R., Pacchiano, A., Stepleton, T., Jiang, H., and Chiappa, S. Wasserstein fair classification. In *Uncertainty in Artificial Intelligence*, pp. 862–872. PMLR, 2020.

Jigsaw. Toxic comment classification challenge, 2018. URL `https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/overview/description`.

Kehrenberg, T., Bartlett, M., Thomas, O., and Quadrianto, N. Null-sampling for interpretable and fair representations. In *European Conference on Computer Vision*, pp. 565–580. Springer, 2020.

Kim, B., Kim, H., Kim, K., Kim, S., and Kim, J. Learning not to learn: Training deep neural networks with biased data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9012–9020, 2019.

Komiyama, J., Takeda, A., Honda, J., and Shimao, H. Nonconvex optimization for regression with fairness constraints. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2737–2746. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/komiyama18a.html`.

Kuleshov, V., Fenner, N., and Ermon, S. Accurate uncertainties for deep learning using calibrated regression. In *International Conference on Machine Learning*, pp. 2796–2804. PMLR, 2018.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436–444, 2015.

Liu, L. T., Simchowitz, M., and Hardt, M. The implicit fairness criterion of unconstrained learning. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4051–4060. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/liu19f.html.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Locatello, F., Abbati, G., Rainforth, T., Bauer, S., Schölkopf, B., and Bachem, O. On the fairness of disentangled representations. *arXiv preprint arXiv:1905.13662*, 2019.

Madras, D., Creager, E., Pitassi, T., and Zemel, R. Learning adversarially fair and transferable representations. In *International Conference on Machine Learning*, pp. 3384–3393. PMLR, 2018.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

NLSY. National longitudinal survey of youth, 2021. URL https://www.bls.gov/nls/.

Obermeyer, Z., Powers, B., Vogeli, C., and Mullainathan, S. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019. doi: 10.1126/science.aax2342. URL https://www.science.org/doi/abs/10.1126/science.aax2342.

Online. Problem with proof of Conditional expectation as best predictor. https://stats.stackexchange.com/questions/71863/problem-with-proof-of-conditional-expectation-as-best-predictor, 2013. [Online; accessed May-2022].

Pedregosa, F. Hyperparameter optimization with approximate gradient. In *International conference on machine learning*, pp. 737–746. PMLR, 2016.

Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. On fairness and calibration. *arXiv preprint arXiv:1709.02012*, 2017.

Raghavan, M., Barocas, S., Kleinberg, J., and Levy, K. Mitigating bias in algorithmic hiring: Evaluating claims and practices. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pp. 469–481, 2020.

Rajeswaran, A., Finn, C., Kakade, S., and Levine, S. Meta-learning with implicit gradients. In *Advances in neural information processing systems*, 2019.

Rosenfeld, E., Ravikumar, P., and Risteski, A. The risks of invariant risk minimization. *arXiv preprint arXiv:2010.05761*, 2020.

Shui, C., Wang, B., and Gagné, C. On the benefits of representation regularization in invariance based domain generalization. *Machine Learning*, pp. 1–21, 2022.

Sjoding, M. W., Dickson, R. P., Iwashyna, T. J., Gay, S. E., and Valley, T. S. Racial bias in pulse oximetry measurement. *New England Journal of Medicine*, 383(25):2477–2478, 2020.

Song, J., Kalluri, P., Grover, A., Zhao, S., and Ermon, S. Learning controllable fair representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2164–2173. PMLR, 2019.

Wightman, L. F. Lsac national longitudinal bar passage study, 1998.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In *International conference on machine learning*, pp. 325–333. PMLR, 2013.

Zhang, B. H., Lemoine, B., and Mitchell, M. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 335–340, 2018.

Zhang, W. and Ntoutsi, E. Faht: an adaptive fairness-aware decision tree classifier. *arXiv preprint arXiv:1907.07237*, 2019.

Zhao, H., Coston, A., Adel, T., and Gordon, G. J. Conditional learning of fair representations. *arXiv preprint arXiv:1910.07162*, 2019.

A. Sufficiency rule: comparison with other fair criteria

Sufficiency v.s. Independence We will demonstrate:

If $\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y)$ (i.e, different label distribution in the sub-groups), the Sufficiency and Independence rule cannot both hold.

Proof. Since we consider the binary-classification with $\mathcal{Y} = \{-1, 1\}$, the expectation and conditional can be expressed as the probability of predicted output.

$$\mathcal{D}_0(\hat{Y} = 1) = \frac{1}{2}(1 + \mathbb{E}_{\mathcal{D}_0}[\hat{Y}]), \;\; \mathcal{D}_0(Y = 1|\hat{Y} = t) = \frac{1}{2}(1 + \mathbb{E}_{\mathcal{D}_0}[Y|\hat{Y} = t])$$

Then sufficiency and independence are equivalent to: $\mathcal{D}_0(\hat{Y} = t) = \mathcal{D}_1(\hat{Y} = t)$ and $\mathcal{D}_0(Y = y|\hat{Y} = t) = \mathcal{D}_1(Y = y|\hat{Y} = t)$ both hold for any t, y. Then the joint distribution of $\hat{Y}, Y$ should be identical:

$$\mathcal{D}_0(\hat{Y} = t, Y = y) = \mathcal{D}_1(\hat{Y} = t, Y = y), \;\; \forall t, y$$

Then the marginal distribution $\mathcal{D}_0(Y = y) = \mathcal{D}_1(Y = y)$ must holds.

If $\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y)$, the joint distribution is not equal:

$$\mathcal{D}_0(\hat{Y} = t, Y = y) \neq \mathcal{D}_1(\hat{Y} = t, Y = y), \;\; \forall t, y$$

Since $\mathcal{D}_0(\hat{Y} = t, Y = y) = \mathcal{D}_0(Y = y|\hat{Y} = t)\mathcal{D}_0(\hat{Y} = t)$, thus either $\mathcal{D}_0(\hat{Y} = t) \neq \mathcal{D}_1(\hat{Y} = t)$ or $\mathcal{D}_0(Y = y|\hat{Y} = t) \neq \mathcal{D}_1(Y = y|\hat{Y} = t)$ must hold for at least one term. I.e, the sufficiency and Independence could not both hold. □

Sufficiency v.s. Separation We will demonstrate:

If $\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y)$ and joint distribution of $(Y, \hat{Y})$ has positive probability in $\mathcal{D}_0, \mathcal{D}_1$, the Sufficiency and Separation rule cannot both hold.

Proof. Based on the previous results, if $\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y)$, then the joint distribution of $(Y, \hat{Y})$ are not identical:

$$\mathcal{D}_0(\hat{Y} = t, Y = y) \neq \mathcal{D}_1(\hat{Y} = t, Y = y), \;\; \forall t, y$$

Then either $\mathcal{D}_0(\hat{Y} = t) \neq \mathcal{D}_1(\hat{Y} = t)$ or $\mathcal{D}_0(Y = y|\hat{Y} = t) \neq \mathcal{D}_1(Y = y|\hat{Y} = t)$ must hold for at least one term (conclusion of previous result).

If the sufficiency and separation both hold when $\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y)$, it must be the following case:

$$\mathcal{D}_0(\hat{Y} = t) \neq \mathcal{D}_1(\hat{Y} = t), \;\; \mathcal{D}_0(Y = y|\hat{Y} = t) = \mathcal{D}_1(Y = y|\hat{Y} = t)$$
$$\mathcal{D}_0(Y = y) \neq \mathcal{D}_1(Y = y), \;\; \mathcal{D}_0(\hat{Y} = t|Y = y) = \mathcal{D}_1(\hat{Y} = t|Y = y)$$

However, we will prove it is impossible in the classification. Based on Bayes rule:

$$\mathcal{D}_0(Y = y|\hat{Y} = t) = \frac{\mathcal{D}_0(\hat{Y} = t|Y = y)\mathcal{D}_0(Y = y)}{\mathcal{D}_0(\hat{Y} = t)} = \frac{\mathcal{D}_1(\hat{Y} = t|Y = y)\mathcal{D}_1(Y = y)}{\mathcal{D}_1(\hat{Y} = t)} = \mathcal{D}_1(Y = y|\hat{Y} = t)$$

Thus we should have

$$\frac{\mathcal{D}_0(Y = y)}{\mathcal{D}_0(\hat{Y} = t)} = \frac{\mathcal{D}_1(Y = y)}{\mathcal{D}_1(\hat{Y} = t)}, \;\; \forall t, y$$

We consider the binary classification by denoting $\mathcal{D}_0(Y = 1) = p, \mathcal{D}_1(y = 1) = q, \mathcal{D}_0(\hat{Y} = 1) = \hat{p}, \mathcal{D}_1(\hat{Y} = 1) = \hat{q}$, then we have:

$$\frac{p}{\hat{p}} = \frac{q}{\hat{q}}, \frac{1-p}{\hat{p}} = \frac{1-q}{\hat{q}}, \frac{p}{1-\hat{p}} = \frac{q}{1-\hat{q}}, \frac{1-p}{1-\hat{p}} = \frac{1-q}{1-\hat{q}}$$

There exists the unique non-zero solution of $p = q = \hat{p} = \hat{q} = 0.5$, which clearly contradicts our assumptions. □

Table 3. Common Fair criteria in classification

	General Definition	Binary classification	Definition	Relation
Independence	$\mathcal{D}_0(\hat{Y}) = \mathcal{D}_1(\hat{Y})$	Demographic parity	$\mathcal{D}_0(\hat{Y} = 1) = \mathcal{D}_1(\hat{Y} = 1)$	Equivalent
Separation	$\mathcal{D}_0(\hat{Y}|Y = y) = \mathcal{D}_1(\hat{Y}|Y = y), \forall y$	Equalized odds	$\mathcal{D}_0(\hat{Y} = 1|Y = y) = \mathcal{D}_1(\hat{Y} = 1|Y = y), \forall y$	Equivalent
Separation		Equal opportunity	$\mathcal{D}_0(\hat{Y} = 1|Y = 1) = \mathcal{D}_1(\hat{Y} = 1|Y = 1)$	Relaxation
Sufficiency	$\mathcal{D}_0(Y|\hat{Y} = y) = \mathcal{D}_1(Y|\hat{Y} = y), \forall y$	Conditional use accuracy equality	$\mathcal{D}_0(Y = y|\hat{Y} = y) = \mathcal{D}_1(Y = y|\hat{Y} = y), \forall y$	Equivalent
Sufficiency		Predictive parity	$\mathcal{D}_0(Y = 1|\hat{Y} = 1) = \mathcal{D}_1(Y = 1|\hat{Y} = 1)$	Relaxation

A.1. Comparison Tables

For the sake of completeness, we list common fair criteria for the comparison.

B. Proof Proposition 3.1

We consider the regression and classification separately.

Regression   According to the definition, given a fixed and deterministic representation λ, we have

$$\mathcal{L}_0(h, \lambda) = \mathbb{E}_{\mathcal{D}_0}(h(z) - y)^2$$

Since it is a functional optimization w.r.t. the function h, through using the calculus of variations (Online, 2013),

$$\frac{\delta \mathcal{L}_0(h, \lambda)}{\delta h(z)} = 2 \int [h(z) - y] \mathcal{D}_0(z, y) dy = 0$$

Solving for $h(z)$, and using the sum and product rules of probability, we obtain

$$h_0^\star(z) = \frac{\int y \mathcal{D}_0(z, y) dy}{\mathcal{D}_0(z)} = \int y \mathcal{D}_0(y|Z = z) dy = \mathbb{E}_{\mathcal{D}_0}[Y|Z = z]$$

Then we have $h_0^\star(z) = \mathbb{E}_{\mathcal{D}_0}[Y|Z = z]$. As for $\mathcal{D}_1$, we apply the same strategy with $h_1^\star(z) = \mathbb{E}_{\mathcal{D}_1}[Y|Z = z]$. Based on the invariant optimal predictor, we have $\mathbb{E}_{\mathcal{D}_0}[Y|Z = z] = \mathbb{E}_{\mathcal{D}_1}[Y|Z = z]$ with $z = \lambda(x)$.

Classification   According to the definition, we have:

$$\mathcal{L}_0(h, \lambda) = \mathbb{E}_{\mathcal{D}_0} \log(1 + \exp(-yh(z)))$$

Since the optimal predictor on the logistic loss is the log-conditional density ratio: $h_0^\star(z) = \log\left(\frac{\mathcal{D}_0(Y = 1|Z = z)}{\mathcal{D}_0(Y = -1|Z = z)}\right)$. Observe that in the binary classification with $Y = \{-1, 1\}$, we have $\mathcal{D}_0(Y = 1|Z = z) = \frac{1}{2}(1 + \mathbb{E}_{\mathcal{D}_0}[Y|Z = z])$ and $\mathcal{D}_0(Y = -1|Z = z) = \frac{1}{2}(1 - \mathbb{E}_{\mathcal{D}_0}[Y|Z = z])$, then we have:

$$h_0^\star(z) = \log\left(\frac{1 + \mathbb{E}_{\mathcal{D}_0}[Y|Z = z]}{1 - \mathbb{E}_{\mathcal{D}_0}[Y|Z = z]}\right)$$

As for $\mathcal{D}_1$, we adopt the same strategy and we have $\log\left(\frac{1 + \mathbb{E}_{\mathcal{D}_0}[Y|Z=z]}{1 - \mathbb{E}_{\mathcal{D}_0}[Y|Z=z]}\right) = \log\left(\frac{1 + \mathbb{E}_{\mathcal{D}_1}[Y|Z=z]}{1 - \mathbb{E}_{\mathcal{D}_1}[Y|Z=z]}\right)$, then we have $\mathbb{E}_{\mathcal{D}_0}[Y|Z = z] = \mathbb{E}_{\mathcal{D}_1}[Y|Z = z]$.

As for the predictive parity, since we have $\mathbb{E}_{\mathcal{D}_0}[Y|Z = z] = \mathbb{E}_{\mathcal{D}_1}[Y|Z = z]$ and $h^\star = h_1^\star = h_2^\star$, then we have $\mathbb{E}_{\mathcal{D}_0}[Y|h^\star(z)] = \mathbb{E}_{\mathcal{D}_1}[Y|h^\star(z)]$.

C. Approximation Error

Theorem C.1 (Approximation Error Gap). *Suppose that (1) **Smooth Predictive Loss**. The first-order derivatives and second-order derivatives of $\mathcal{L}$ are Lipschitz continuous; (2) **Non-singular Hessian matrix**. We assume*

$\nabla_{h_0,h_0}\mathcal{L}_0(h_0, \lambda), \nabla_{h_1,h_1}\mathcal{L}_1(h_1, \lambda)$, *the Hessian matrix of the inner optimization problem, are invertible. (3) **Bounded representation and predictor function**. We assume the λ and h are bounded, i.e., $\|\lambda\|, \|h\|$ are upper bounded by the predefined positive constants. Then the approximation error between the ground truth and algorithmic estimated gradient w.r.t. the representation is be upper bounded by:*

$$\|grad(\lambda) - \tilde{grad}^{\delta}(\lambda)\| = \mathcal{O}(\kappa\epsilon + \epsilon + \delta).$$

Proof. We denote grad(λ) as the ground truth gradient w.r.t. λ in outer-loop loss (given the optimal predictor $h_0^\star$, $h_1^\star$). Then we aim to bound

$$\|\text{grad}(\lambda) - \tilde{\text{grad}}^{\delta}(\lambda)\|$$

We first introduce the following terms for facilitating the proof:

$$A_0^\epsilon = \nabla_{h_0}\nabla_\lambda\mathcal{L}_0(h_0^\epsilon, \lambda), A_1^\epsilon = \nabla_\lambda\nabla_{h_1}\mathcal{L}_1(h_1^\epsilon, \lambda), A_0^\star = \nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^\star, \lambda), A_1^\star = \nabla_\lambda\nabla_{h_1}\mathcal{L}_1(h_1^\star, \lambda),$$
$$B_0^\epsilon = \nabla_\lambda\mathcal{L}_0(h_0^\epsilon, \lambda), B_1^\epsilon = \nabla_\lambda\mathcal{L}_1(h_1^\epsilon, \lambda), B_0^\star = \nabla_\lambda\mathcal{L}_0(h_0^\star, \lambda), B_1^\star = \nabla_\lambda\mathcal{L}_1(h_1^\star, \lambda),$$
$$\mathbf{p}_0^\star = \left(\nabla_{h_0}^2\mathcal{L}_0(h_0^\star, \lambda)\right)^{-1}\left(\nabla_{h_0}\mathcal{L}_0(h_0^\star, \lambda) + \kappa(h_0^\star - h_1^\star)\right),$$
$$\mathbf{p}_1^\star = \left(\nabla_{h_1}^2\mathcal{L}_1(h_1^\star, \lambda)\right)^{-1}\left(\nabla_{h_1}\mathcal{L}_1(h_1^\star, \lambda) - \kappa(h_0^\star - h_1^\star)\right).$$

Then the approximation error gap can be expressed as:

$$\|\text{grad}(\lambda) - \tilde{\text{grad}}^{\delta}(\lambda)\| = \|\left(B_0^\star - A_0^\star\mathbf{p}_0^\star + B_1^\star - A_1^\star\mathbf{p}_1^\star\right) - \left(B_0^\epsilon - A_0^\epsilon\mathbf{p}_0^\delta + B_1^\epsilon - A_1^\epsilon\mathbf{p}_1^\delta\right)\|$$
$$\leq \sum_{i=0}^{1}\|B_i^\star - B_i^\epsilon\| + \sum_{i=0}^{1}\|A_i^\star\mathbf{p}_i^\star - A_i^\delta\mathbf{p}_i^\delta\|$$

Due to the symmetric of $\mathcal{D}_0$ and $\mathcal{D}_1$, we only focus on the term on $i = 0$, the the upper bound in $i = 1$ can be derived analogously.

As for bounding $\|B_0^\star - B_0^\epsilon\|$, since we assume first order derivative of the loss is Lipschitz functions (with constant L_1), then we have :

$$\|B_0^\star - B_0^\epsilon\| \leq L_1\|h_0^\star - h_0^\epsilon\| \leq \epsilon L_1$$

Then the second term can be upper bounded by three terms:

$$\|A_0^\star\mathbf{p}_0^\star - A_0^\delta\mathbf{p}_0^\delta\| \leq \underbrace{\|A_0^\star\mathbf{p}_0^\star - A_0^\star\mathbf{p}_0\|}_{(1)} + \underbrace{\|A_0^\star\mathbf{p}_0 - A_0^\epsilon\mathbf{p}_0\|}_{(2)} + \underbrace{\|A_0^\epsilon\mathbf{p}_0 - A_0^\epsilon\mathbf{p}_0^\delta\|}_{(3)}$$

Before estimating the upper bound, we first demonstrate $\|A_0^\epsilon\|$ and $\|A_0^\star\|$ are also bounded.

Since we assume λ and h are bounded (assuming the bounded constant as η and ϕ), the second order derivative are Lipschitz (with constant L_2). Then we consider another fixed point $(\lambda', h_0^\star(\lambda'))$ with bounded second order derivative: $A_0 = \nabla_{h_0,\lambda}^2\mathcal{L}_0(h_0^\star(\lambda'), \lambda')$ and $\|A_0\| \leq A$. We have:

$$\|A_0^\star - A_0\|_2 \leq L_2\|[h_0^\star(\lambda), \lambda] - [h_0^\star(\lambda'), \lambda']\|_2 \leq L_2\sqrt{\eta^2 + \phi^2}$$

Thus we have $\|A_0^\star\| \leq A + L_2\sqrt{\eta^2 + \phi^2} = A_{\text{sup}}^\star$. As for the second derivative at point h_0^ϵ, it can be upper bounded analogously with a similar constant A_{sup}^ϵ.

The upper bound of term (1) We have:

$$\|A_0^\star\mathbf{p}_0^\star - A_0^\star\mathbf{p}_0\| \leq \|A_0^\star\|\|\mathbf{p}_0^\star - \mathbf{p}_0\|$$

We have proved $\|A_0^\star\|$ is upper bounded by $A_{\text{sup}}^\star$. We additionally introduce the following auxiliary terms:

$$P_0^\star = \left(\nabla_{h_0}^2\mathcal{L}_0(h_0^\star, \lambda)\right)^{-1}, P_0^\epsilon = \left(\nabla_{h_1}^2\mathcal{L}_1(h_1^\star, \lambda)\right)^{-1}.$$
$$b_0^\star = \nabla_{h_0}\mathcal{L}_0(h_0^\star, \lambda) + \kappa(h_0^\star - h_1^\star), b_0^\epsilon = \nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda) + \kappa(h_0^\epsilon - h_1^\epsilon)$$

Then we have:

$$\|\mathbf{p}_0^\star - \mathbf{p}_0\| = \|P_0^\star b_0^\star - P_0^\epsilon b_0^\epsilon\|$$
$$\leq \|P_0^\star b_0^\star - P_0^\star b_0^\epsilon\| + \|P_0^\star b_0^\epsilon - P_0^\epsilon b_0^\epsilon\|$$
$$\leq \|P_0^\star\|\|b_0^\star - b_0^\epsilon\| + \|b_0^\epsilon\|\|P_0^\star - P_0^\epsilon\|$$

As for the $\|P_0^\star\|$, since we assume the Hessian matrix is invertible thus its norm is upper bounded by some constant (denoted as A_{-1}). As for $\|b_0^\star - b_0^\epsilon\|$, we have:

$$\|b_0^\star - b_0^\epsilon\| \leq \|\nabla_{h_0}\mathcal{L}_0(h_0^\star, \lambda) - \nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda)\| + 2\kappa\epsilon$$
$$\leq \epsilon L_1 + 2\kappa\epsilon$$

Thus we have $\|P_0^\star\|\|b_0^\star - b_0^\epsilon\| \leq A_{-1}(\epsilon L_1 + 2\kappa\epsilon)$.

As for $\|b_0^\epsilon\|$, we can easily verify that it is indeed bounded by some constant b. For the first term, we can adopt the same strategy in proving bounded $\|A_0^\star\|$. As for the second term in b_0^ϵ, it is upper bounded by $2\kappa\phi$, due to the bounded predictor.

We now demonstrate $\|P_0^\star - P_0^\epsilon\|$. Denoting $\Delta = (P_0^\star)^{-1} - (P_0^\epsilon)^{-1}$, then according to the second order Lipschitz assumption, we have: $\|\Delta\| \leq \epsilon L_2$. Plugging in the result, we have:

$$\|P_0^\star - P_0^\epsilon\| = \|(P_0^\star)\Delta(P_0^\epsilon)\| \leq \|P_0^\star\|\|\Delta\|\|P_0^\epsilon\| \leq (A_{-1})^2 L_2\epsilon$$

We still adopt the assumption that the bounded Hessian-inverse matrix by A_{-1}.

Plugging in all the results, we have:

$$(1) \leq A_1(\epsilon L_1 + 2\kappa\epsilon) + b(A_1)^2 L_2\epsilon := \mathcal{O}(\kappa\epsilon + \epsilon)$$

The upper bound of term (2)   We have:

$$\|A_0^\star \mathbf{p}_0 - A_0^\epsilon \mathbf{p}_0\| \leq \|\mathbf{p}_0\|_2 \|A_0^\star - A_0^\epsilon\|$$

Since we assume the loss is second-order Lipschitz, thus we have

$$\|A_0^\star - A_0^\epsilon\| = \|\nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^\star, \lambda) - \nabla_\lambda\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda)\| \leq L_2\|h_0^\star - h_0^\epsilon\| \leq \epsilon L_2$$

We can also demonstrate $\|\mathbf{p}_0\|$ is bounded. According to the definition we have:

$$\|\mathbf{p}_0\| \leq \|\left(\nabla_{h_0}^2 \mathcal{L}_0(h_0^\epsilon, \lambda)\right)^{-1}\|\|\left(\nabla_{h_0}\mathcal{L}_0(h_0^\epsilon, \lambda) + \kappa(h_0^\epsilon - h_1^\epsilon)\right)\|$$
$$\overset{(i)}{\leq} A_{-1}(L_1\|h_0^\star - h_0^\epsilon\|_2 + 2\kappa\phi)$$
$$\overset{(ii)}{\leq} A_{-1}(\epsilon L_1 + 2\kappa\phi)$$

For (i), we assume: 1) the Hessian matrix is invertible thus its norm is surely upper bounded by some constant (denoted as A_{-1}), 2) the first-order derivative is Lipschitz (bounded by L_1), 3) the predictor h is bounded. For (ii), we adopt the definition of h_0^ϵ.

Therefore, the upper bound for Term (2) is formulated as:

$$(2) \leq \epsilon L_2 A_{-1}(\epsilon L_1 + 2\kappa\phi) := \mathcal{O}(\kappa\epsilon)$$

The upper bound of term (3)   We have:

$$\|A_0^\epsilon \mathbf{p}_0 - A_0^\epsilon \mathbf{p}_0^\delta\| \leq \|A_0^\epsilon\|\|\mathbf{p}_0 - \mathbf{p}_0^\delta\| \leq \delta A_{\sup}^\epsilon = \mathcal{O}(\delta)$$

Through the upper bound in (1)-(3), we finally have the error between the estimated and ground-truth gradient:

$$\|\text{grad}(\lambda) - \tilde{\text{grad}}^\delta(\lambda)\| = \mathcal{O}(\kappa\epsilon + \epsilon + \delta)$$

$\square$

D. The Convergence Behavior

For the sake of completeness, we provide the convergence analysis of the proposed algorithm.

Proposition D.1. *We execute the implicit alignment algorithm (Algo. 1), obtaining a sequence of $\lambda_1, \ldots, \lambda_k, \ldots$. Supposing the fair constraint κ is fixed. The optimization tolerances are summable: $\sum_k \epsilon_k^2 \leq +\infty$ and $\sum_k \delta_k^2 \leq +\infty$, then λ_k is proved to be converged with*

$$\lim_{k \to \infty} \lambda_k = \lambda^\star.$$

If the stationary point $\lambda^\star$ is also within the bounded norm, then we have:

$$grad(\lambda^\star) = 0.$$

Proof. We denote the entire outer-loop loss w.r.t. λ as $\mathcal{L}(\lambda)$, by the assumption the β-smooth loss $\mathcal{L}$. Then at iteration $k+1$ and k, we have:

$$\mathcal{L}(\lambda_{k+1}) \leq \mathcal{L}(\lambda_k) - \text{grad}(\lambda_k)^T (\lambda_k - \lambda_{k+1}) + \frac{\beta}{2} \|\lambda_{k+1} - \lambda_k\|^2$$

$$= \mathcal{L}(\lambda_k) - \left(\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k) + \tilde{\text{grad}}^\delta(\lambda_k) \right)^T (\lambda_k - \lambda_{k+1}) + \frac{\beta}{2} \|\lambda_{k+1} - \lambda_k\|^2$$

$$= \mathcal{L}(\lambda_k) - \left(\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k) \right)^T (\lambda_k - \lambda_{k+1}) - \tilde{\text{grad}}^\delta(\lambda_k)(\lambda_k - \lambda_{k+1}) + \frac{\beta}{2} \|\lambda_{k+1} - \lambda_k\|^2$$

Since we assume the representation is within the bounded norm, the projection onto the convex set are non-expansive operators (Boyd et al., 2004). Then for any point p, q, we have $\|\text{proj}(p) - \text{proj}(q)\|^2 \leq (p - q)^T (\text{proj}(p) - \text{proj}(q))$. Then we set λ_k and $\lambda_{k+1} = \lambda_k - \frac{1}{\beta} \tilde{\text{grad}}^\delta(\lambda_k)$, we have:

$$\|\lambda_k - \lambda_{k+1}\|^2 \leq \frac{1}{\beta} (\tilde{\text{grad}}^\delta(\lambda_k))^T (\lambda_k - \lambda_{k+1})$$

Plugging into the results, we have:

$$\mathcal{L}(\lambda_{k+1}) \leq \mathcal{L}(\lambda_k) - \left(\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k) \right)^T (\lambda_k - \lambda_{k+1}) - \frac{\beta}{2} \|\lambda_{k+1} - \lambda_k\|^2$$

$$\leq \mathcal{L}(\lambda_k) + \|\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k)\| \|\lambda_k - \lambda_{k+1}\| - \frac{\beta}{2} \|\lambda_{k+1} - \lambda_k\|^2$$

Rearranging the inequality, we have:

$$\frac{\beta}{2} \|\lambda_{k+1} - \lambda_k\|^2 - \|\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k)\| \|\lambda_k - \lambda_{k+1}\| + (\mathcal{L}(\lambda_{k+1}) - \mathcal{L}(\lambda_k)) \leq 0$$

Then we have:

$$\|\lambda_{k+1} - \lambda_k\| \leq \frac{1}{\beta} \left(\|\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k)\| + \sqrt{\|\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k)\|^2 - 2\beta \left(\mathcal{L}(\lambda_{k+1}) - \mathcal{L}(\lambda_k) \right)} \right)$$

By denoting $B_k = \|\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k)\|$ and $C_k = \mathcal{L}(\lambda_{k+1}) - \mathcal{L}(\lambda_k)$. Then we have:

$$\|\lambda_{k+1} - \lambda_k\|^2 \leq \frac{1}{\beta^2} \left(B_k^2 + B_k^2 - 2\beta C_k + 2 B_k \sqrt{B_k^2 - 2\beta C_k} \right)$$

$$\leq \frac{1}{\beta^2} \left(B_k^2 + B_k^2 - 2\beta C_k + B_k^2 + B_k^2 - 2\beta C_k \right)$$

$$= \frac{4}{\beta^2} [\|\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k)\|_2^2 - 2\beta \left(\mathcal{L}(\lambda_{k+1}) - \mathcal{L}(\lambda_k) \right)]$$

Taking sum over k, we have:

$$\sum_{k=1}^{+\infty} \|\lambda_{k+1} - \lambda_k\|^2 \leq \frac{4}{\beta^2} \sum_{k=1}^{+\infty} \|\text{grad}(\lambda_k) - \tilde{\text{grad}}^\delta(\lambda_k)\|_2^2 - \frac{8}{\beta}(\lim_{k\to\infty} \mathcal{L}(\lambda_{k+1}) - \mathcal{L}(\lambda_1))$$

$$\leq \frac{4}{\beta^2} \sum_k [(C+\kappa)^2 \epsilon_k^2 + \delta_k^2] - \frac{8}{\beta}\left(\lim_{k\to\infty} \mathcal{L}(\lambda_{k+1}) - \mathcal{L}(\lambda_1)\right) < +\infty$$

Since 1) the first term on the right side is finite, because the optimization tolerance is summable; 2) the second term is also finite, because the loss is assumed to be bounded. Then the upper bound is finite. In order to satisfy this condition, on the left side we should have:

$$\lim_{k\to\infty} \lambda_{k+1} - \lambda_k = 0$$

By adopting the definition $\lambda_{k+1} = \text{Proj}(\lambda_k - \tilde{\text{grad}}^\delta(\lambda_k))$ and $\lim_{k\to\infty} \tilde{\text{grad}}^\delta(\lambda_k) = \text{grad}(\lambda_k)$ (Based on theorem 1, the limit of the optimization tolerance is zero), then we have:

$$\lambda^\star = \text{proj}(\lambda^\star - \text{grad}(\lambda^\star))$$

Where $\lambda^\star = \lim_{k\to+\infty} \lambda_{k+1} = \lim_{k\to+\infty} \lambda_k$. Since the projection is on the bounded norm L_{norm} and $\lambda^\star$ is within the bounded norm space, thus if $\lambda^\star - \text{grad}(\lambda^\star)$ is within the bounded norm space, we have:

$$\text{grad}(\lambda^\star) = 0$$

Else if $\lambda^\star - \text{grad}(\lambda^\star)$ is outside the bounded norm space, then according to the definition, the projection of $\lambda^\star - \text{grad}(\lambda^\star)$ is surely on the *boundary* of the L_{norm} space, with $\|\text{proj}(\lambda^\star - \text{grad}(\lambda^\star))\| = L_{\text{norm}}$. However, we have assumed the $\lambda^\star$ is *within* the bounded norm space with $\|\lambda^\star\| < L_{\text{norm}}$, which leads to the contradiction. Based on these discussions, we finally have:

$$\text{grad}(\lambda^\star) = 0$$

$\square$

E. Possible extensions to non-binary protected features

For the completeness, it is also possible to extend to binary protected features with distribution $D_1, \ldots, D_N$. For example, the bi-level objective can be naturally formulated as

$$\min_\lambda \sum_{n=1}^N \mathcal{L}_n(h_n^\star, \lambda) + \sum_{m=1, n=1, n<m}^{n=N, m=N} \frac{\kappa_{n,m}}{2} \|h_n^\star - h_m^\star\|_2^2$$

$$\text{s.t.} \qquad h_n^\star \in \text{argmin}_h \mathcal{L}_n(h, \lambda)$$

Compared with the binary group, we introduce the pair-wise regularization ($\|h_n^\star - h_m^\star\|$) term to ensure the invariance between each pair of the sensitive attributes (n, m). However, determining the coefficient $\kappa_{n,m}$ will become practically challenging, since the hyper-parameter space is much larger $O(N^2)$ than the binary case.

If the sensitive attribute is indeed a real value, a simple practical approach is to cluster the continuous attribute into several discrete groups, then conducting the pair-wise bi-level optimization. At the same time, there may be difficulty in measuring fairness. E.g, if the sensitive attribute is the ratio of people of certain demographic backgrounds, the corresponding sufficiency gap will be hard to estimate, since current metrics are defined on the discrete sensitive attribute.

F. Additional Details and Results

F.1. Illustrative example of sufficiency gap

F.2. Correlation Analysis on the benchmark

For the justification propose, we compute the Pearson correlation coefficient (ranging from $[-1, 1]$) between the binary group index (or protected feature) A and label Y. Intuitively, if $\mathcal{D}_{A=a}(Y = y) = \mathcal{D}_{A=a'}(Y = y), \forall a, a', y$, the protected feature (or group index) A is independent of label Y.

Table 4. Pearson correlation coefficient between the group index and label

Toxic-Comment	CelebA	Law	NLSY
0.30	-0.35	0.18	-0.29

The Pearson correlation coefficient clearly demonstrates the *non-independence* between the group index and label. i.e, the label distributions among the sub-groups are different. The experimental results also validated this fact: *the demographic parity based approach could not improve the sufficiency gap* due to the different label distributions.

F.3. Additional Details

Toxic Comments We split the training, validation and testing set as 70%, 10% and 20%. We adopt Adam optimizer with learning rate 10^{-3} and eps 10^{-3}. The batch-size is set as 500 for each sub-group and we use sampling with replacement to run the explicit algorithm with maximum epoch 100. The fair coefficient is generally set as $\kappa = 0.1 \sim 0.001$. As for the inner-optimization step, the iteration number is 20 and the iteration in running conjugate gradient approach is 10.

CelebA The training/validation/test set are around 82K, 18K and 18K. We also adopt the Adam optimizer with learning rate on $\lambda : 10^{-5} \sim 10^{-4}$ and $h : 10^{-3}$. The batch-size is set as 64 for each sub-group and we iterate the whole dataset as one epoch. The maximum running epoch is set as 20 and the iteration in running conjugate gradient approach is 10.

Law We split the training, validation and testing set as 70%, 10% and 20%. Then we adopt Adam optimizer with learning rate 10^{-3} and eps 10^{-3}. The batch-size is set as 500 for each sub-group and we use sampling with replacement to run the implicit algorithm, with the maximum epoch 100. We adopt the MSE loss in the regression. The fair coefficient is generally set as $\kappa = 0.1 \sim 10^{-4}$. As for the inner-optimization, the iteration number is 20 and the iteration in running conjugate gradient is 10. In computing the sufficiency gap in the regression, we sample 33 points to compute the gap.

NLSY We split the training, validation and testing set as 70%, 10% and 20%. Then we adopt Adam optimizer with learning rate 10^{-3} and eps 10^{-3}. The batch-size is set as 500 for each sub-group and we use sampling with replacement to run the implicit algorithm, with maximum epoch 100. We adopt the MSE loss in the regression. The fair coefficient is generally set as $\kappa = 0.1 \sim 10^{-4}$. As for the inner-optimization, the iteration number is 20 and the iteration in running conjugate gradient is 10. In computing the sufficiency gap, we sample 33 points to compute the sufficiency gap.
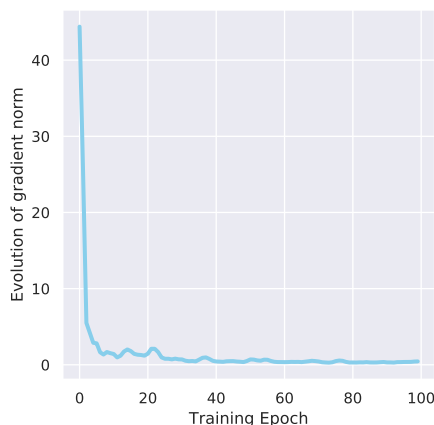
F.4. Additional Empirical results

Figure 7. Gradient Norm evolution w.r.t. representation λ in Toxic comments dataset. We visualize the norm of $\tilde{\text{grad}}^\delta(\lambda)$ at each training epoch, which suggests a convergence behavior and the gradient finally tends to zero.

Gradient evolution We also visualize the gradient norm of the representation λ in the Toxic dataset, shown in Fig. 7. The results verify the convergence behavior and the gradient norm finally tends to zero.

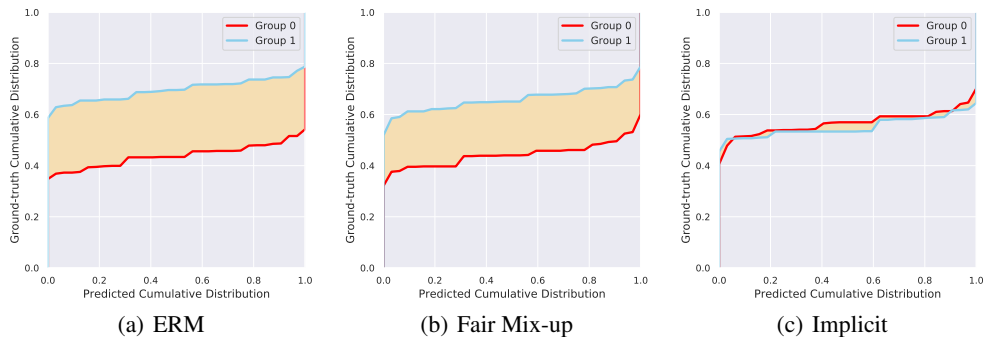| (a) ERM	(b) Fair Mix-up	(c) Implicit

Figure 8. Illustration of the *sufficiency gap* in NLSY dataset. The ERM and mix-up suffer the high predictive sufficiency-gap, while the proposed implicit alignment can significantly mitigate the sufficiency gap. In contrast, the probability calibration is not improved. This results also verifies the inequivalence between the sufficiency gap and calibration gap (Liu et al., 2019).

F.5. Discussion with non-deep learning baselines

In order to show the effectiveness of the proposed approach, we additionally compare the FAHT (Zhang & Ntoutsi, 2019), a decision tree based fair classification approach. We evaluated the empirical performance on Toxic comments dataset.

Table 5. Comparison with Fairness Aware Decision Tree

Method	Accuracy ($\uparrow$)	$\Delta\mathrm{Suf}_C$ ($\downarrow$)
FAHT	0.596	0.397
Implicit	0.760	0.051

The implicit approach demonstrates the considerable better results, which may come from two aspects: (1) the Toxic task is a high-dimensional classification problem ($x \in \mathbb{R}^{748}$), where the deep learning based approach is more effective in handling the high-dim dataset. (2) The FAHT aims to realize the statistical parity (the independence rule), which is *not compatible* with the sufficiency. According to the analysis of (Barocas et al., 2019), when the protected feature (A) and label (Y) are not independent (This has been justified by computing their Pearson Correlation coefficient), the sufficiency and independence cannot both hold.

F.6. sufficiency Gap in regression

We visualize the sufficiency gap of NLSY dataset.

G. Complementary technical details

We present complementary details that are related to the paper.

G.1. Conjugate Gradient Method

We present the Conjugate Gradient (CG) algorithm in Algo. 2 through `autograd`. In the conventional CG algorithm with objective $\frac{1}{2}x^T A X - bX$, we need to estimate AX and compute its residual and update X. Since in our problem setting, the $A = \nabla^2_{h_0} \mathcal{L}_0(h_0^\epsilon, \lambda)$, then computing AX can be realized through Hessian-vector product through `autograd`, denoted as function F in the paper. i.e., $\nabla^2_{h_0} \mathcal{L}_0(h_0^\epsilon, \lambda)X = F(x)$.

Below we provided a simple PyTorch code for realizing the Hessian Vector product.

```
import torch
def hessain_vector_product(loss,model,vector):
    # loss: the defined loss
    # model: the model in computing the Hessian
    # vector: the required vector in computing Hessian-vector product
    partial_grad = torch.autograd.grad(loss, model_parameters(), create_graph=True)
```

Algorithm 2 Conjugate Gradient Method

Ensure: Function F that computes Hessian-vector product through `autograd`, initial value X_0, bias vector B.

1: Computing Residual: $r_0 = B - F(X_0)$
2: Set $p_0 = r_0$
3: **for** inner_iterations k **do**
4:     Computing $\alpha_k \leftarrow \frac{r_k^T r_k}{p_k^T F(p_k)}$
5:     $X_{k+1} \leftarrow X_k + \alpha_k p_k$
6:     $r_{k+1} \leftarrow r_k - \alpha_k F(p_k)$
7:     If r_{k+1} is sufficiency small, then stop.
8:     $\beta_k \leftarrow \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
9:     $p_{k+1} \leftarrow r_{k+1} + \beta_k p_k$
10: **end for**
11: **Return:** X_{k+1}

```
7     flat_grad = torch.cat([g.contiguous().view(-1) for g in partial_grad])
8     h = torch.sum(flat_grad * vector_to_optimize)
9     hvp = torch.autograd.grad(h, model.parameters())
10    return hvp
```

Listing 1. Simple demo in computing Hessian vector product

G.2. Calibration Gap in the regression

Based on Kuleshov et al. (2018), we first compute the predicted cumulative distribution $(\hat{Y}_0)$ of at point t: $D_0(\hat{Y}_0 \leq t) = \alpha$, then we compute the corresponding ground truth cumulative distribution (Y_0) at point t. By changing t, we obtain several points on function $\mathcal{D}_0(Y \leq t | \hat{Y}_0 \leq t) = \beta$. Then the regression is probabilistic calibrated when $\alpha \equiv \beta$. From this perspective, the zero calibration gap can guarantee a zero sufficiency gap. But the inverse is not necessarily true, as our experimental results suggest, a small sufficiency gap can lead to either small or large calibration gap. Thus it can be quite promising to explore their inherent relations and trade-off in the fair regression.