Imparting Fairness to Pre-Trained Biased Representations

Bashir Sadeghi
Michigan State University
sadeghib@msu.edu

Vishnu Naresh Boddeti
Michigan State University
vishnu@msu.edu

Abstract

Adversarial representation learning is a promising paradigm for obtaining data representations that are invariant to certain sensitive attributes while retaining the information necessary for predicting target attributes. Existing approaches solve this problem through iterative adversarial minimax optimization and lack theoretical guarantees. In this paper, we first study the "linear" form of this problem i.e., the setting where all the players are linear functions. We show that the resulting optimization problem is both non-convex and non-differentiable. We obtain an exact closed-form expression for its global optima through spectral learning. We then extend this solution and analysis to non-linear functions through kernel representation. Numerical experiments on UCI and CIFAR-100 datasets indicate that, (a) practically, our solution is ideal for "imparting" provable invariance to any biased pre-trained data representation, and (b) empirically, the trade-off between utility and invariance provided by our solution is comparable to iterative minimax optimization of existing deep neural network based approaches.

Code is available at Human Analysis Lab.

1. Introduction

Adversarial representation learning (ARL) is a promising framework for training image representation models that can control the information encapsulated within it. ARL is practically employed to learn representations for a variety of applications, including, unsupervised domain adaptation of images [9], censoring sensitive information from images [8], learning fair and unbiased representations [18, 19], learning representations that are controllably invariant to sensitive attributes [28] and mitigating unintended information leakage [24], amongst others.

At the core of the ARL formulation is the idea of jointly optimizing three entities: (i) An encoder E that seeks to distill the information from input data and retains the information relevant to a target task while *intentionally* and *permanently* eliminating the information corresponding to a sensitive attribute, (ii) A predictor T that seeks to extract a desired target attribute, and (iii) A proxy adversary A, playing the role of an unknown adversary, that seeks to extract a known sensitive attribute. Figure 1 shows a pictorial illustration of the ARL problem.

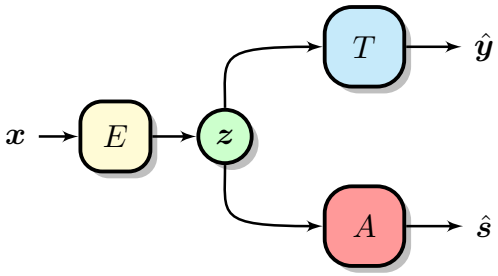


Figure 1: **Adversarial Representation Learning** consists of three entities, an encoder E that obtains a compact representation $\mathbf{z}$ of input data $\mathbf{x}$, a predictor T that predicts a desired target attribute $\mathbf{y}$ and an adversary that seeks to extract a sensitive attribute $\mathbf{s}$, both from the embedding $\mathbf{z}$.

Typical instantiations of ARL represent these entities through non-linear functions in the form of deep neural networks (DNNs) and formulate parameter learning as a minimax optimization problem. Practically, optimization is performed through simultaneous gradient descent, wherein, small gradient steps are taken concurrently in the parameter space of the encoder, predictor and proxy adversary. The solutions thus obtained have been effective in learning data representations with controlled invariance across applications such as image classification [24], multi-lingual machine translation [28] and domain adaptation [9].

Despite its practical promise, the aforementioned ARL setup suffers from a number of drawbacks:

– The minimax formulation of ARL leads to an optimization problem that is non-convex in the parameter space, both due to the adversarial loss function as well as due to the non-linear nature of modern DNNs. As we show in this paper, even for simple instances of ARL where each entity is characterized by a linear function, the problem remains non-convex in the parameter space. Similar observations [23] have been

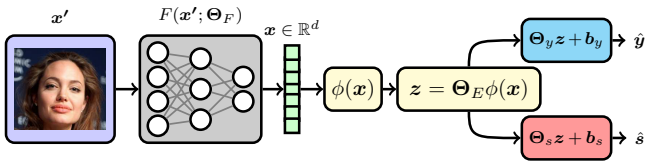


Figure 2: **Overview:** Illustration of adversarial representation learning for imparting invariance to a fixed biased pre-trained image representation $\mathbf{x} = \mathbf{F}(\mathbf{x}'; \mathbf{\Theta_F})$. An encoder E, in the form of a kernel mapping, produces a new representation $\mathbf{z}$. A target predictor and an adversary, in the form of linear regressors, operate on this new representation. We theoretically analyze this ARL setup to obtain a closed form solution for the globally optimal parameters of the encoder $\mathbf{\Theta_E}$. Provable bounds on the achievable trade-off between the utility and fairness of the representation are also derived.

made in a different but related context of adversarial learning in generative adversarial networks (GANs) [11].

– In applications of ARL related to fairness, accountability and transparency of machine learning models, it is critically important to provide performance bounds in addition to empirical evidence of model efficacy. A major shortcoming of existing DNN based ARL solutions is the lack of theoretical analysis or provable bounds on achievable utility and fairness.

In this paper, we take a step back and analytically study the simplest version of the ARL problem from an optimization perspective with the goal of addressing the aforementioned drawbacks. Doing so enables us to delineate the contributions of the expressivity of the entities in ARL (i.e., shallow vs deep models) and the challenges of optimizing the parameters (i.e., local optima through simultaneous gradient descent vs global optima).

Contributions: We first consider the "linear" form of ARL, where the encoder is a linear transformation, the target predictor is a linear regressor and proxy adversary is a linear regressor. We show that this Linear-ARL leads to an optimization problem that is both non-convex and non-differentiable. Despite this fact, by reducing it into a set of trace problems on a Stiefel manifold, we obtain an exact closed form solution for the global optima. As part of our solution, we also determine optimal dimensionality of the embedding space. Finally, we extend the Linear-ARL formulation to allow non-linear functions through a kernel extension while still enjoying an exact closed-form solution for the global optima. Numerical experiments on multiple datasets, both small and large scale, indicate that the global optima solution for the linear and kernel formulations of ARL are competitive and sometimes even outperform DNN based ARL trained through simultaneous stochastic gradient descent. Practically, we also demonstrate the utility of Linear-ARL and Kernel-ARL for "imparting" provable invariance to any biased pre-trained data representation. Figure 2 provides an overview of our contributions. We refer to our proposed algorithm for obtaining the global optima as Spectral-ARL and abbreviate it as SARL.

Notation: Scalars are denoted by regular lower case or Greek letters, *e.g.* n, λ. Vectors are denoted by boldface lowercase letters, *e.g.* $\mathbf{x}$, $\mathbf{y}$. Matrices are uppercase boldface letters, *e.g.* $\mathbf{X}$. A $k \times k$ identity matrix is denoted by $\mathbf{I}_k$ or $\mathbf{I}$. Centered (mean subtracted w.r.t. columns) data matrix is indicated by "~", *e.g.* $\tilde{\mathbf{X}}$. Assume that $\mathbf{X}$ contains n columns, then $\tilde{\mathbf{X}} = \mathbf{XD}$, where $\mathbf{D} = \mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ and $\mathbf{1}$ denotes the vector of ones with length of n. Given matrix $\mathbf{M} \in \mathbb{R}^{m \times m}$, we use $\mathrm{Tr}[\mathbf{M}]$ to denote its trace (i.e., the sum of its diagonal elements); its Frobenius norm is denoted by $\|\mathbf{M}\|_F$, which is related to the trace as $\|\mathbf{M}\|_F^2 = \mathrm{Tr}[\mathbf{M}\mathbf{M}^T] = \mathrm{Tr}[\mathbf{M}^T\mathbf{M}]$. The subspace spanned by the columns of $\mathbf{M}$ is denoted by $\mathcal{R}(\mathbf{M})$ or simply $\mathcal{M}$ (in calligraphy); the orthogonal complement of $\mathcal{M}$ is denoted by $\mathcal{M}^\perp$. The null space of $\mathbf{M}$ is denoted by $\mathbb{N}(\mathbf{M})$. The orthogonal projection onto $\mathcal{M}$ is $P_{\mathcal{M}} = \mathbf{M}(\mathbf{M}^T\mathbf{M})^\dagger\mathbf{M}^T$, where superscript "$\dagger$" indicates the Moore-Penrose pseudo inverse [17].

Let $\mathbf{x} \in \mathbb{R}^d$ be a random vector. We denote its expectation by $\mathbb{E}[\mathbf{x}]$, and its covariance matrix by $\mathbf{C}_x \in \mathbb{R}^{d \times d}$ as $\mathbf{C}_x = \mathbb{E}\big[(\mathbf{x} - \mathbb{E}[\mathbf{x}])(\mathbf{x} - \mathbb{E}[\mathbf{x}])^T\big]$. Similarly, the cross-covariance $\mathbf{C}_{xy} \in \mathbb{R}^{d \times r}$ between $\mathbf{x} \in \mathbb{R}^d$ and $\mathbf{y} \in \mathbb{R}^r$ is denoted as $\mathbf{C}_{xy} = \mathbb{E}\big[(\mathbf{x} - \mathbb{E}[\mathbf{x}])(\mathbf{y} - \mathbb{E}[\mathbf{y}])^T\big]$.

For a $d \times d$ positive definite matrix $\mathbf{C} \succ 0$, its Cholesky factorization results in a full rank matrix $\mathbf{Q} \in \mathbb{R}^{d \times d}$ such that

$$\mathbf{C} = \mathbf{Q}^T\mathbf{Q} \tag{1}$$

2. Prior Work

Adversarial Representation Learning: In the context of image classification, adversarial learning has been utilized to obtain representations that are invariant across domains [9, 10, 27]. Such representations allow classifiers that are trained on a source domain to generalize to a different target domain. In the context of learning fair and unbiased representations, a number of approaches [1, 3, 8, 21, 24, 28, 30] have used and argued [19] for explicit adversarial networks[1], to extract sensitive attributes from the encoded data. With the exception of [24] all the other methods are set up as a minimax game between the encoder, a target task and the adversary. The encoder is setup to achieve fairness by maximizing the loss of the adversary i.e. minimizing negative log-likelihood of sensitive variables as measured by the adversary. Roy *et al.* [24] identify and address the instability in the optimization in the zero-sum minimax formulation of

[1] Proxies at training to mimic unknown adversaries during inference.

ARL and propose an alternate non-zero sum solution, demonstrating significantly improved empirical performance. All the above approaches use deep neural networks to represent the ARL entities, optimize their parameters through simultaneous stochastic gradient descent, and rely on empirical validation. However, none of them seek to study the nature of the ARL formulation itself i.e., in terms of decoupling the role of the expressiveness of the models and convergence/stability properties of the optimization tools for learning the parameters of said models. Therefore, we seek to bridge this gap by studying simpler forms of ARL from a global optimization perspective.

Privacy, Fairness and Invariance: Concurrent work on learning fair or invariant representations of data included an encoder and a target predictor but did not involve an explicit adversary. The role of the adversary is played by an explicit hand designed objective that, typically, competes with that of the target task. The concept of learning fair representations was first introduced by Zemel *et al*. [29]. The goal was to learn a representation of data by "fair clustering" while maintaining the discriminative features of the prediction task. Building upon this work, many techniques have been proposed to learn an unbiased representation of data while retaining its effectiveness for a prediction task. These include the Variational Fair Autoencoder [18] and the more recent information bottleneck based objective by Moyer *et al*. [22]. As with the ARL methods above, these approaches rely on empirical validation. Neither of them study their respective non-convex objectives from an optimization perspective, nor do they provide any provable guarantees on achievable trade-off between fairness and utility. The competing nature of the objectives considered in this body of work shares resemblance to the non-convex objectives that we study in this paper. Though it is not our focus, the approach presented here could potentially be extended to analyze the aforementioned methods.

Optimization Theory for Adversarial Learning: The non-convex nature of the ARL formulation poses unique challenges from an optimization perspective. Practically, the parameters of the models in ARL are optimized through stochastic gradient descent, either jointly [8, 20] or alternatively [9], with the former being a generalization of gradient descent. While the convergence properties of gradient descent and its variants are well understood, there is relatively little work on the convergence and stability of simultaneous gradient descent in adversarial minimax problems. Recently, Mescheder *et al*. [20] and Nagarajan *et al*. [23] both leveraged tools from non-linear systems theory [12] to analyze the convergence properties of simultaneous gradient descent, in the context of GANs, around a given equilibrium. They show that without the introduction of additional regularization terms to the objective of the zero-sum game, simultaneous gradient descent does not converge. However, their analysis is restricted to the two player GAN setting and is not concerned with its global optima.

In the context of fair representation learning, Komiyama *et al*. [15] consider the problem of enforcing fairness constraints in linear regression and provide a solution to obtain the global optima of the resulting non-convex problem. While we derive inspiration from this work, our problem setting and technical solution are both notably different. Specifically, their approach does not involve, (1) an explicit adversary as a measure of sensitive information in the representation, and (2) an encoder tasked with disentangling and discarding the sensitive information in the data.

3. Adversarial Representation Learning

Let the data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ be n realizations of d-dimensional data $\mathbf{x} \in \mathbb{R}^d$. Assume that $\mathbf{x}$ is associated with a sensitive attribute $\mathbf{s} \in \mathbb{R}^q$ and a target attribute $\mathbf{y} \in \mathbb{R}^p$. We denote n realizations of sensitive and target attributes as $\mathbf{S} = [\mathbf{s}_1, \cdots, \mathbf{s}_n]$ and $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_n]$, respectively. Treating the attributes as vectors enables us to consider both multi-class classification and regression under the same setup.

3.1. Problem Setting

The adversarial representation learning problem is formulated with the goal of learning parameters of an embedding function $E(\cdot; \mathbf{\Theta}_E) : \mathbf{x} \mapsto \mathbf{z}$ with two objectives: (i) aiding a target predictor $T(\cdot; \mathbf{\Theta}_y)$ to accurately infer the target attribute $\mathbf{y}$ from $\mathbf{z}$, and (ii) preventing an adversary $A(\cdot; \mathbf{\Theta}_s)$ from inferring the sensitive attribute $\mathbf{s}$ from $\mathbf{z}$. The ARL problem can be formulated as,

$$\begin{aligned} \min_{\mathbf{\Theta}_E} \min_{\mathbf{\Theta}_y} \; & \mathcal{L}_y \left(T(E(\mathbf{x}; \mathbf{\Theta}_E); \mathbf{\Theta}_y), \mathbf{y} \right) \\ \text{s.t.} \; & \min_{\mathbf{\Theta}_s} \; \mathcal{L}_s \left(A(E(\mathbf{x}; \mathbf{\Theta}_E); \mathbf{\Theta}_s), \mathbf{s} \right) \geq \alpha \end{aligned} \tag{2}$$

where $\mathcal{L}_y$ and $\mathcal{L}_s$ are the loss functions (averaged over training dataset) for the target predictor and the adversary, respectively, $\alpha \in [0, \infty)$ is a user defined value that determines the minimum tolerable loss for the adversary on the sensitive attribute, and the minimization in the constraint is equivalent to the encoder operating against an optimal adversary. Existing instances of this problem adopt deep neural networks to represent E, T and A and learn their respective parameters $\{\mathbf{\Theta}_E, \mathbf{\Theta}_y, \mathbf{\Theta}_s\}$ through simultaneous SGD.

3.2. The Linear Case

We first consider the simplest form of the ARL problem and analyze it from an optimization perspective. We model both the adversary and the target predictors as linear regressors,

$$\hat{\mathbf{y}} = \mathbf{\Theta}_y \mathbf{z} + \mathbf{b}_y, \qquad \hat{\mathbf{s}} = \mathbf{\Theta}_s \mathbf{z} + \mathbf{b}_s \tag{3}$$

where $\mathbf{z}$ is an encoded version of $\mathbf{x}$, and $\hat{\mathbf{y}}$ and $\hat{\mathbf{s}}$ are the predictions corresponding to the target and sensitive attributes. We also model the encoder through a linear mapping,

$$\boldsymbol{\Theta}_E \in \mathbb{R}^{r\times d} \quad : \quad \mathbf{x} \mapsto \mathbf{z} = \boldsymbol{\Theta}_E \mathbf{x} \tag{4}$$

where $r < d$ is the dimensionality[2] of the projected space. While existing DNN based solutions select r on an ad-hoc basis, our approach for this problem determines r as part of our solution to the ARL problem. For both adversary and target predictors, we adopt the mean squared error (MSE) to assess the quality of their respective predictions i.e., $\mathcal{L}_y(\mathbf{y}, \hat{\mathbf{y}}) = \mathbb{E}[\|\mathbf{y} - \hat{\mathbf{y}}\|^2]$ and $\mathcal{L}_s(\mathbf{s}, \hat{\mathbf{s}}) = \mathbb{E}[\|\mathbf{s} - \hat{\mathbf{s}}\|^2]$.

3.2.1 Optimization Problem

For any given encoder $\boldsymbol{\Theta}_E$ the following Lemma gives the minimum MSE for a linear regressor in terms of covariance matrices and $\boldsymbol{\Theta}_E$. The following Lemma assumes that $\mathbf{x}$ is zero-mean and the covariance matrix $\mathbf{C}_x$ is positive definite. These assumptions are not restrictive since we can always remove the mean and dependent features from $\mathbf{x}$.

Lemma 1. *Let $\mathbf{x}$ and $\mathbf{t}$ be two random vectors with $\mathbb{E}[\mathbf{x}] = 0$, $\mathbb{E}[\mathbf{t}] = \mathbf{b}$, and $\mathbf{C}_x \succ 0$. Consider a linear regressor, $\hat{\mathbf{t}} = \mathbf{W}\mathbf{z} + \mathbf{b}$, where $\mathbf{W} \in \mathbb{R}^{m\times r}$ is the parameter matrix, and $\mathbf{z} \in \mathbb{R}^r$ is an encoded version of $\mathbf{x}$ for a given $\boldsymbol{\Theta}_E$: $\mathbf{x} \mapsto \mathbf{z} = \boldsymbol{\Theta}_E\mathbf{x}$, $\boldsymbol{\Theta}_E \in \mathbb{R}^{r\times d}$. The minimum MSE that can be achieved by designing $\mathbf{W}$ is,*

$$\min_{\mathbf{W}} \mathbb{E}[\|\mathbf{t} - \hat{\mathbf{t}}\|^2] = \mathrm{Tr}\big[\mathbf{C}_\mathbf{t}\big] - \big\|\mathbf{P}_\mathcal{M}\mathbf{Q}_\mathbf{x}^{-\mathbf{T}}\mathbf{C}_\mathbf{xt}\big\|_\mathbf{F}^\mathbf{2}$$

where $\mathbf{M} = \mathbf{Q}_x\boldsymbol{\Theta}_E^T \in \mathbb{R}^{d\times r}$, and $\mathbf{Q}_x \in \mathbb{R}^{d\times d}$ is a Cholesky factor of $\mathbf{C}_x$ as shown in (1).

Applying this result to the target and adversary regressors, we obtain their minimum MSEs,

$$\begin{aligned} J_y(\boldsymbol{\Theta}_E) &= \min_{\boldsymbol{\Theta}_y} \mathcal{L}_y\left(T(E(\mathbf{x};\boldsymbol{\Theta}_E);\boldsymbol{\Theta}_y), \mathbf{y}\right) \\ &= \mathrm{Tr}\big[\mathbf{C}_y\big] - \big\|P_\mathcal{M}\mathbf{Q}_x^{-T}\mathbf{C}_{xy}\big\|_F^2 \end{aligned} \tag{5}$$

$$\begin{aligned} J_s(\boldsymbol{\Theta}_E) &= \min_{\boldsymbol{\Theta}_s} \mathcal{L}_s\left(A(E(\mathbf{x};\boldsymbol{\Theta}_E);\boldsymbol{\Theta}_s), \mathbf{s}\right) \\ &= \mathrm{Tr}\big[\mathbf{C}_s\big] - \big\|P_\mathcal{M}\mathbf{Q}_x^{-T}\mathbf{C}_{xs}\big\|_F^2 \end{aligned} \tag{6}$$

Given the encoder, $J_y(\boldsymbol{\Theta}_E)$ is related to the performance of the target predictor; whereas $J_s(\boldsymbol{\Theta}_E)$ corresponds to the amount of sensitive information that an adversary is able to leak. Note that the linear model for T and A enables us to obtain their respective optimal solutions for a given encoder $\boldsymbol{\Theta}_E$. On the other hand, when T and A are modeled as DNNs, doing the same is analytically infeasible and potentially impractical.

[2]When r is equal to d, the encoder will be unable to guard against the adversary who can simply learn to invert $\boldsymbol{\Theta}_E$.

The orthogonal projector $P_\mathcal{M}$ in Lemma 1 is a function of two factors, a data dependent term $\mathbf{Q}_x$ and the encoder parameters $\boldsymbol{\Theta}_E$. While the former is fixed for a given dataset, the latter is our object of interest. Pursuantly, we decompose $P_\mathcal{M}$ in order to separably characterize the effect of these two factors. Let the columns of $\mathbf{L}_x \in \mathbb{R}^{d\times d}$ be an orthonormal basis for the column space of $\mathbf{Q}_x$. Due to the bijection $\mathbf{G}_E = \mathbf{L}_x^{-1}\mathbf{Q}_x\boldsymbol{\Theta}_E^T \Leftrightarrow \boldsymbol{\Theta}_E = \mathbf{G}_E^T\mathbf{L}_x^T\mathbf{Q}_x^{-T}$ from $\mathbf{L}_x\mathbf{G}_E = \mathbf{Q}_x\boldsymbol{\Theta}_E^T$, determining the encoder parameters $\boldsymbol{\Theta}_E$ is equivalent to determining $\mathbf{G}_E$. The projector $P_\mathcal{M}$ can now be expressed in terms of $P_\mathcal{G}$, which is only dependent on the free parameter $\mathbf{G}_E$.

$$P_\mathcal{M} = \mathbf{M}\big(\mathbf{M}^T\mathbf{M}\big)^\dagger\mathbf{M}^T = \mathbf{L}_x P_\mathcal{G}\mathbf{L}_x^T \tag{7}$$

where we used the equality $\mathbf{M} = \mathbf{Q}_x\boldsymbol{\Theta}_E^T$ and the fact that $\mathbf{L}_x^T\mathbf{L}_x = \mathbf{I}$.

Now, we turn back to the ARL setup and see how the above decomposition can be leveraged. The optimization problem in (2) reduces to,

$$\begin{aligned} &\min_{\mathbf{G}_E} \; J_y(\mathbf{G}_E) \\ &\text{s.t.} \; J_s(\mathbf{G}_E) \geq \alpha \end{aligned} \tag{8}$$

where the minimum MSE measures of (5) and (6) are now expressed in terms of $\mathbf{G}_E$ instead of $\boldsymbol{\Theta}_E$.

Before solving this optimization problem, we will first interpret it geometrically. Consider a simple example where $\mathbf{x}$ is a white random vector i.e., $\mathbf{C}_x = \mathbf{I}$. Under this setting, $\mathbf{Q}_x = \mathbf{L}_x = \mathbf{I}$ and $\mathbf{G}_E = \boldsymbol{\Theta}_E^T$. As a result, the optimization problem in (8) can alternatively be solved in terms of $\mathbf{G}_E = \boldsymbol{\Theta}_E^T$ as $J_y(\mathbf{G}_E) = \mathrm{Tr}\big[\mathbf{C}_y\big] - \big\|P_\mathcal{G}\mathbf{C}_{xy}\big\|_F^2$ and $J_s(\mathbf{G}_E) = \mathrm{Tr}\big[\mathbf{C}_s\big] - \big\|P_\mathcal{G}\mathbf{C}_{xs}\big\|_F^2$.

The constraint $J_s(\mathbf{G}_E) \geq \alpha$ implies $\big\|P_\mathcal{G}\mathbf{C}_{xs}\big\|_F^2 \leq \big(\mathrm{Tr}\big[\mathbf{C}_s\big] - \alpha\big)$ which is geometrically equivalent to the subspace $\mathcal{G}$ being outside (or tangent to) the cone around $\mathbf{C}_{xs}$. Similarly, minimizing $J_y(\mathbf{G}_E)$ implies maximizing $\big\|P_\mathcal{G}\mathbf{C}_{xy}\big\|_F^2$, which in turn is equivalent to minimizing the angle between the subspace $\mathcal{G}$ and the vector $\mathbf{C}_{xy}$. Therefore, the global optima of (8) is any hyper plane $\mathcal{G}$ which is outside the cone around $\mathbf{C}_{xs}$ while subtending the smallest angle to $\mathbf{C}_{xy}$. An illustration of this setting and its solution is shown in Figure 3 for $d = 3$, $r = 2$ and $p = q = 1$.

Constrained optimization problems such as (8) are commonly solved through their respective unconstrained Lagrangian [2] formulations as shown below

$$\min_{\mathbf{G}_E \in \mathbb{R}^{d\times r}} \Big\{(1-\lambda)J_y(\mathbf{G}_E) - (\lambda)J_s(\mathbf{G}_E)\Big\} \tag{9}$$

for some parameter $0 \leq \lambda \leq 1$. Such an approach affords two main advantages and one disadvantage; (a) A direct and closed-form solution can be obtained. (b) Framing (9) in terms of λ and $(1-\lambda)$ allows explicit control between the two extremes of *no privacy* ($\lambda = 0$) and *no target*

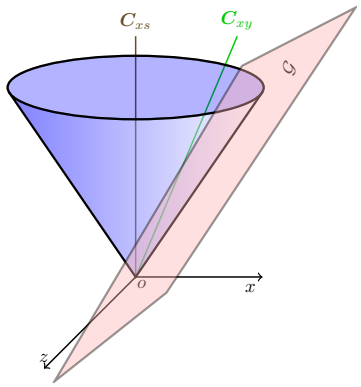


Figure 3: **Geometric Interpretation:** An illustration of a three-dimensional input space $\mathbf{x}$ and one-dimensional target and adversary regressors. Therefore, both $\mathbf{C}_{xs}$ and $\mathbf{C}_{xy}$ are one-dimensional. We locate the y-axis in the same direction as $\mathbf{C}_{xs}$. The feasible space for the solution $\mathbf{G}_E = \mathbf{\Theta}_E^T$ imposed by the constraint $J_s(\mathbf{\Theta}_E) \geq \alpha$ corresponds to the region *outside* the cone (specified by $\mathbf{C}_s$ and α) around $\mathbf{C}_{xs}$. The non-convexity of the problem stems from the non-convexity of this feasible set. The objective $\min J_y(\mathbf{\Theta}_E)$ corresponds to minimizing the angle between the line $\mathbf{C}_{xy}$ and the plane $\mathcal{G}$. When $\mathbf{C}_{xy}$ is outside the cone, the line $\mathbf{C}_{xy}$ itself or any plane that contains the line $\mathbf{C}_{xy}$ and does not intersect with the cone, is a valid solution. When $\mathbf{C}_{xy}$ is inside the cone, the solution is either a line or, as we illustrate, a tangent hyperplane to the cone that is closest to $\mathbf{C}_{xy}$. The non-differentiability stems from the fact that the solution can either be a plane or a line.

($\lambda = 1$). As a consequence, it can be shown that for every $\lambda \in [0, 1]$, $\exists\, \alpha \in [\alpha_{\min}, \alpha_{\max}]$. In practice, given a user specified value of $\alpha_{\min} \leq \alpha_{\text{tol}} \leq \alpha_{\max}$, we can solve (8) by iterating over $\lambda \in [0, 1]$ until the solution of (9) yields the same specified α_{tol}. (c) The vice-versa on the other hand does not necessarily hold i.e., for a given tolerable loss α there may not be a corresponding $\lambda \in [0, 1]$. This is the theoretical limitation of solving Lagrangian problem instead of the constrained problem.

Before we obtain the solution to the Lagrangian formulation (9), we characterize the nature of the optimization problem in the following theorem.

Theorem 2. *As a function of $\mathbf{G}_E \in \mathbb{R}^{d\times r}$, the objective function in (9) is neither convex nor differentiable.*

3.2.2 Learning

Despite the difficulty associated with the objective in (9), we derive a closed-form solution for its global optima. Our key insight lies in partitioning the search space $\mathbb{R}^{d\times r}$ based on the rank of the matrix $\mathbf{G}_E$. For a given rank i, let $\mathcal{S}_i$ be the set containing all matrices $\mathbf{G}_E$ of rank i,

$$\mathcal{S}_i = \big\{\mathbf{G}_E \in \mathbb{R}^{d\times r} \;\big|\; \text{rank}(\mathbf{G}_E) = i\big\}, \quad i = 0, 1, \cdots, r$$

Obviously, $\bigcup_{i=0}^{r} \mathcal{S}_i = \mathbb{R}^{d\times r}$. As a result, the optimization problem in (9) can be solved by considering r minimization problems, one for each possible rank of $\mathbf{G}_E$:

$$\min_{i\in\{1,\ldots,r\}} \Big\{ \min_{\mathbf{G}_E\in\mathcal{S}_i} (1-\lambda) J_y(\mathbf{G}_E) - (\lambda) J_s(\mathbf{G}_E) \Big\} \quad (10)$$

We observe from (5), (6) and (7), that the optimization problem in (9) is dependent only on a subspace $\mathcal{G}$. As such, the solution $\mathbf{G}_E$ is not unique since many different matrices can span the same subspace. Hence, it is sufficient to solve for any particular $\mathbf{G}_E$ that spans the optimal subspace $\mathcal{G}$. Without loss of generality we seek an orthonormal basis spanning the optimal subspace $\mathcal{G}$ as our desired solution. We constrain $\mathbf{G}_E \in \mathbb{R}^{d\times i}$ to be an orthonormal matrix i.e., $\mathbf{G}_E^T\mathbf{G}_E = \mathbf{I}_i$ where i is the dimensionality of $\mathcal{G}$. Ignoring the constant terms in J_y and J_s, for each $i = 1, \ldots, r$, the minimization problem over $\mathcal{S}_i$ in (10) reduces to,

$$\min_{\mathbf{G}_E^T\mathbf{G}_E=\mathbf{I}_i} J_\lambda(\mathbf{G}_E) \quad (11)$$

where

$$\begin{aligned} J_\lambda(\mathbf{G}_E) = \lambda &\|\mathbf{L}_x\mathbf{G}_E\mathbf{G}_E^T\mathbf{L}_x^T\mathbf{Q}_x^{-T}\mathbf{C}_{xs}\|_F^2 \\ &- (1-\lambda)\|\mathbf{L}_x\mathbf{G}_E\mathbf{G}_E^T\mathbf{L}_x^T\mathbf{Q}_x^{-T}\mathbf{C}_{xy}\|_F^2 \end{aligned}$$

From basic properties of trace, we have, $J_\lambda(\mathbf{G}_E) = \text{Tr}\big[\mathbf{G}_E^T\mathbf{B}\mathbf{G}_E\big]$ where $\mathbf{B} \in \mathbb{R}^{d\times d}$ is a symmetric matrix:

$$\mathbf{B} = \mathbf{L}_x^T\mathbf{Q}_x^{-T}\big(\lambda\,\mathbf{C}_{sx}^T\mathbf{C}_{sx} - (1-\lambda)\,\mathbf{C}_{yx}^T\mathbf{C}_{yx}\big)\mathbf{Q}_x^{-1}\mathbf{L}_x \quad (12)$$

The optimization problem in (11) is equivalent to trace minimization on a Stiefel manifold which has closed-form solution(s) (see [14] and [7]).

In view of the above discussion the solution to the optimization problem in (9) or equivalently (10) can be stated in the next theorem.

Theorem 3. *Assume that the number of negative eigenvalues (β) of $\mathbf{B}$ in (12) is j. Denote $\gamma = \min\{r, j\}$. Then, the minimum value in (10) is given as,*

$$\beta_1 + \beta_2 + \cdots + \beta_\gamma \quad (13)$$

where $\beta_1 \leq \beta_2 \leq \ldots \leq \beta_\gamma < 0$ are the γ smallest eigenvalues of $\mathbf{B}$. And the minimum can be attained by $\mathbf{G}_E = \mathbf{V}$, where the columns of $\mathbf{V}$ are eigenvectors corresponding to all the γ negative eigenvalues of $\mathbf{B}$.

Note that, including the eigenvectors corresponding to zero eigenvalues of $\mathbf{B}$ into our solution $\mathbf{G}_E$ in Theorem 3

does not change the minimum value in (13). But, considering only negative eigenvectors results in $\mathbf{G}_E$ with the least rank and thereby an encoder that is less likely to contain sensitive information for an adversary to exploit. Once $\mathbf{G}_E$ is constructed, we can obtain our desired encoder as, $\mathbf{\Theta}_E = \mathbf{G}_E^T \mathbf{L}_x^T \mathbf{Q}_x^{-T}$. Recall that the solution in Theorem 3 is under the assumption that the covariance $\mathbf{C}_x$ is a full-rank matrix.

3.3. Non-Linear Extension Through Kernelization

We extend the "linear" version of the ARL problem studied thus far to a "non-linear" version through kernelization. We model the encoder in the ARL problem as a linear function over non-linear mapping of inputs as illustrated in Figure 2. Let the data matrix $\mathbf{X}$ be mapped non-linearly by a possibly unknown and infinite dimensional function $\phi_x(\cdot)$ to $\mathbf{\Phi}_x$. Let the corresponding reproducing kernel function be $k_x(\cdot,\cdot)$. The centered kernel matrix can be obtained as,

$$\tilde{\mathbf{K}}_x = \tilde{\mathbf{\Phi}}_x^T \tilde{\mathbf{\Phi}}_x = \mathbf{D}^T \mathbf{\Phi}_x^T \mathbf{\Phi}_x \mathbf{D} = \mathbf{D}^T \mathbf{K}_x \mathbf{D} \tag{14}$$

where $\mathbf{K}_x$ is the kernel matrix on the original data $\mathbf{X}$.

If the co-domain of $\phi_x(\cdot)$ is infinite dimensional (e.g., RBF kernel), then the encoder in (4) would be also be infinite dimensional i.e., $\mathbf{\Theta}_E \in \mathbb{R}^{r \times \infty}$, which is infeasible to learn directly. However, the representer theorem [26] allows us to construct the encoder as a linear function of $\tilde{\mathbf{\Phi}}^T$, i.e, $\mathbf{\Theta}_E = \mathbf{\Lambda} \tilde{\mathbf{\Phi}}_x^T = \mathbf{\Lambda} \mathbf{D}^T \mathbf{\Phi}_x^T$. Hence, a data sample $\mathbf{x}$ can be mapped through the "kernel trick" as,

$$\begin{aligned} \mathbf{\Theta}_E \phi_x(\mathbf{x}) &= \mathbf{\Lambda} \mathbf{D}^T \mathbf{\Phi}_x^T \phi_x(x) \\ &= \mathbf{\Lambda} \mathbf{D}^T [k_x(\mathbf{x}_1, \mathbf{x}), \cdots, k_x(\mathbf{x}_n, \mathbf{x})]^T \end{aligned} \tag{15}$$

Hence, designing $\mathbf{\Theta}_E$ is equivalent to designing $\mathbf{\Lambda} \in \mathbb{R}^{r \times n}$. The Lagrangian formulation of this Kernel-ARL setup and its solution shares the same form as that of the linear case (9). The objective function remains non-convex and non-differentiable, while the matrix $\mathbf{B}$ is now dependent on the kernel matrix $\mathbf{K}_x$ as opposed to the covariance matrix $\mathbf{C}_x$

$$\mathbf{B} = \mathbf{L}_x^T \big(\lambda \tilde{\mathbf{S}}^T \tilde{\mathbf{S}} - (1-\lambda) \tilde{\mathbf{Y}}^T \tilde{\mathbf{Y}}\big) \mathbf{L}_x \tag{16}$$

where the columns of $\mathbf{L}_x$ are the orthonormal basis for $\tilde{\mathbf{K}}_x$. Once $\mathbf{G}_E$ is obtained through the eigendecomposition of $\mathbf{B}$, we can find $\mathbf{\Lambda}$ as $\mathbf{\Lambda} = \mathbf{G}_E^T \mathbf{L}_x^T \tilde{\mathbf{K}}_x^{\dagger}$. This non-linear extension in the form of kernelization serves to study the ARL problem under a setting where the encoder possess greater representational capacity while still being able to obtain the global optima and bounds on objectives of the target predictor and the adversary as we show next.

4. Numerical Experiments

We evaluate the efficacy of the proposed Spectral-ARL (SARL) algorithm in finding the global optima, and compare it with other ARL baselines that are based on the standard simultaneous SGD optimization (henceforth referred to as SSGD). In all experiments we refer to our solution for "linear" ARL as Linear-SARL and the solution to the "kernel" version of the encoder with linear classifiers for the predictor and adversary as Kernel-SARL.

Table 1: Fair Classification Performance (in %)

	Adult Dataset			German Dataset		
Method	Target (income)	Sensitive (gender)	Δ^*	Target (credit)	Sensitive (age)	Δ^*
Raw Data	85.0	85.0	17.6	80.0	87.0	6.0
LFR [29]	82.3	67.0	0.4	72.3	80.5	0.5
VAE [13]	81.9	66.0	1.4	72.5	79.5	1.5
VFAE [18]	81.3	67.0	0.4	72.7	79.7	1.3
ML-ARL [28]	84.4	67.7	0.3	74.4	80.2	0.8
MaxEnt-ARL [24]	84.6	65.5	1.9	72.5	80.0	1.0
Linear-SARL	84.1	67.4	0.0	76.3	80.9	0.1
Kernel-SARL	84.1	67.4	0.0	76.3	80.9	0.1

* Absolute difference between adversary accuracy and random guess

4.1. Fair Classification

We consider the task of learning representations that are invariant to a sensitive attribute on two datasets, Adult and German, from the UCI ML-repository [6]. For comparison, apart from the raw features $\mathbf{X}$, we consider several baselines that use DNNs and trained through simultaneous SGD; LFR [29], VAE [13], VFAE [18], ML-ARL [28] and MaxEnt-ARL [24].

The Adult dataset contains 14 attributes. There are $30,163$ and $15,060$ instances in the training and test sets, respectively. The target task is binary classification of annual income i.e., more or less than 50K and the sensitive attribute is gender. Similarly, the German dataset contains 1000 instances of individuals with 20 different attributes. The target is to classify the credit of individuals as good or bad with the sensitive attribute being age.

We learn encoders on the training set, after which, following the baselines, we freeze the encoder and train the target (logistic regression) and adversary (2 layer network with 64 units) classifiers on the training set. Table 1 shows the performance of target and adversary on both datasets. Both Linear-SARL and Kernel-SARL outperform all DNN based baselines. For either of these tasks, the Kernel-SARL does not afford any additional benefit over Linear-SARL. For the adult dataset, the linear encoder maps the 14 input features to just one dimension. The weights assigned to each feature is shown in Figure 4. Notice that the encoder assigns almost zero weight to the gender feature in order to be fair with respect to the gender attribute.

4.2. CIFAR-100

The CIFAR-100 dataset [16] consists of 50,000 images from 100 classes that are further grouped into 20 super-

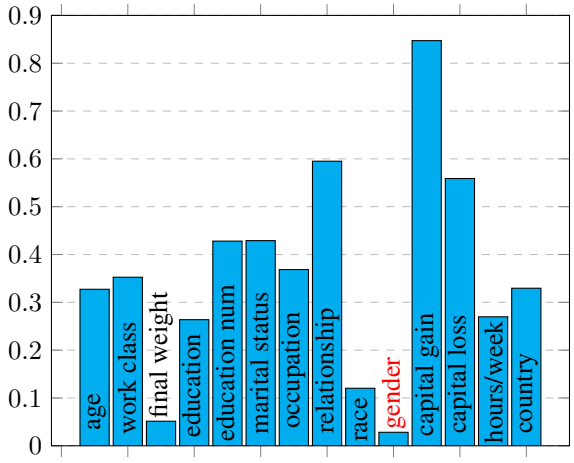
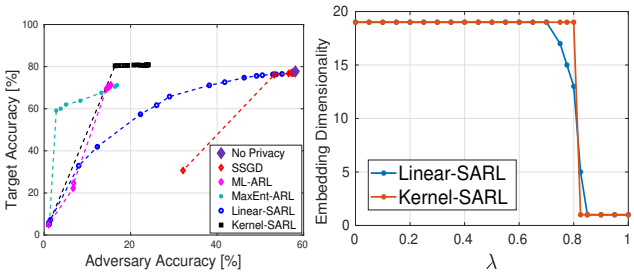


Figure 4: **Adult Dataset:** Magnitude of learned encoder weights $\mathbf{\Theta_E}$ for each semantic input feature.



(a) Trade-off (b) Embedding Dimensionality

Figure 5: **CIFAR-100:** (a) Trade-off between target performance and leakage of sensitive attribute by adversary. (b) Optimal embedding dimensionality learned by SARL. At small values of λ, the objective favors the target task which predicts 20 classes. Thus, embedding dimensionality of 19 is optimal for a linear target regressor. At large values of λ, the objective only seeks to hinder the adversary. Thus, SARL determines the optimal dimensionality of the embedding as one.

classes. Each image is therefore associated with two attributes, a "fine" class label and a "coarse" superclass label. We consider a setup where the "coarse" and "fine" labels are the target and sensitive attributes, respectively. For Linear-SARL and Kernel-SARL (degree five polynomial kernel) and SSGD we use features (64-dimensional) extracted from a pre-trained ResNet-110 model as an input to the encoder, instead of raw images. From these features, the encoder is tasked with aiding the target predictor and hindering the adversary. This setup serves as an example to illustrate how invariance can be "imparted" to an existing biased pre-trained representation. We also consider two DNN baselines, ML-ARL [28] and MaxEnt-ARL [24]. Unlike our scenario, where the pre-trained layers of ResNet-18 are not adapted, the baselines optimize the entire encoder for the ARL task. For evaluation, once the encoder is learned and frozen, we train a discriminator and adversary as 2-layer networks with 64 neurons each. Therefore, although our approach uses linear regressor as adversary at training, we evaluate against stronger adversaries at test time. In contrast, the baselines train and evaluate against adversaries with equal capacity.

Figure 5a shows the trade-off in accuracy between the target predictor and adversary. We observe that, (1) Kernel-ARL significantly outperforms Linear-SARL. Since the former implicitly maps the data into an higher dimensional space, the sensitive features are potentially disentangled sufficiently for the linear encoder in that space to discard such information. Therefore, even for large values of λ, Kernel-SARL is able to simultaneously achieve high target accuracy while keeping the adversary performance low. (2) Despite being handicapped by the fact that Kernel-SARL is evaluated against stronger adversaries than it is trained against, its performance is comparable to that of the DNN baselines. In fact, it outperforms both ML-ARL and MaxEnt-ARL with respect to the target task. (3) Despite repeated attempts with different hyper-parameters and choice of optimizers, SSGD was highly unstable across most datasets and often got stuck in a local optima and failed to find good solutions.

Figure 5b plots the optimal embedding dimensionality provided by SARL as a function of the trade-off parameter λ. At small values of λ, the objective favors the target task i.e., 20 class prediction. Thus, SARL does indeed determine the optimal dimensionality of 19 for a 20 class linear target regressor. However, at large values of λ, the objective only seeks to hinder the sensitive task i.e., 100 class prediction. In this case, the ideal embedding dimensionality from the perspective of the linear adversary regressor is at least 99. The SARL ascertained dimensionality of one is, thus, optimal for maximally mitigating the leakage of sensitive attribute from the embedding. However, unsurprisingly, the target task also suffers significantly.

5. Concluding Remarks

We studied the "linear" form of adversarial representation learning (ARL), where all the entities are linear functions. We showed that the optimization problem even for this simplified version is both non-convex and non-differentiable. Using tools from spectral learning we obtained a closed form expression for the global optima and derived analytical bounds on the achievable utility and invariance. We also extended these results to non-linear parameterizations through kernelization. Numerical experiments on multiple datasets indicated that the global optima solution of the "kernel" form of ARL is able to obtain a trade-off between utility and invariance that is comparable to that of local optima solutions

of deep neural network based ARL.

Admittedly, the results presented in this paper do not extend directly to deep neural network based formulations of ARL. However, we believe it sheds light on nature of the ARL optimization problem and aids our understanding of the ARL problem. It helps delineate the role of the optimization algorithm and the choice of embedding function, highlighting the trade-off between the expressivity of the functions and our ability to obtain the global optima of the adversarial game. We consider our contribution as the first step towards controlling the non-convexity that naturally appears in game-theoretic representation learning.

References

[1] Martin Bertran, Natalia Martinez, Afroditi Papadaki, Qiang Qiu, Miguel Rodrigues, Galen Reeves, and Guillermo Sapiro. Adversarially learned representations for information obfuscation and inference. In *International Conference on Machine Learning*, 2019. 2

[2] DP Bertsekas. Nonlinear programming 2nd edn (belmont, ma: Athena scientific). 1999. 4

[3] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075*, 2017. 2

[4] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[5] Saheb Chhabra, Richa Singh, Mayank Vatsa, and Gaurav Gupta. Anonymizing k-facial attributes via adversarial perturbations. *arXiv preprint arXiv:1805.09380*, 2018.

[6] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. 6

[7] Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1998. 5

[8] Harrison Edwards and Amos Storkey. Censoring representations with an adversary. *arXiv preprint arXiv:1511.05897*, 2015. 1, 2, 3

[9] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, 2015. 1, 2, 3

[10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 2

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014. 2

[12] Hassan K Khalil. Nonlinear systems. *Prentice Hall*, 1996. 3

[13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 6

[14] Effrosini Kokiopoulou, Jie Chen, and Yousef Saad. Trace optimization and eigenproblems in dimension reduction methods. *Numerical Linear Algebra with Applications*, 18(3):565–602, 2011. 5

[15] Junpei Komiyama, Akiko Takeda, Junya Honda, and Hajime Shimao. Nonconvex optimization for regression with fairness constraints. In *International Conference on Machine Learning*, 2018. 3

[16] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 6

[17] Alan J Laub. *Matrix analysis for scientists and engineers*, volume 91. SIAM, 2005. 2

[18] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830*, 2015. 1, 3, 6

[19] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. In *International Conference on Machine Learning*, 2018. 1, 2

[20] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. In *Advances in Neural Information Processing Systems*, 2017. 3

[21] Vahid Mirjalili, Sebastian Raschka, Anoop Namboodiri, and Arun Ross. Semi-adversarial networks: Convolutional autoencoders for imparting privacy to face images. In *International Conference on Biometrics*, 2018. 2

[22] Daniel Moyer, Shuyang Gao, Rob Brekelmans, Aram Galstyan, and Greg Ver Steeg. Invariant representations without adversarial training. In *Advances in Neural Information Processing Systems*, 2018. 3

[23] Vaishnavh Nagarajan and J Zico Kolter. Gradient descent gan optimization is locally stable. In *Advances in Neural Information Processing Systems*, 2017. 1, 3

[24] Proteek Roy and Vishnu Naresh Boddeti. Mitigating information leakage in image representations: A maximum entropy approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1, 2, 6, 7

[25] Bashir Sadeghi, Runyi Yu, and Vishnu Boddeti. On the global optima of kernelized adversarial representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7971–7979, 2019.

[26] John Shawe-Taylor, Nello Cristianini, et al. *Kernel methods for pattern analysis*. Cambridge University Press, 2004. 6

[27] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[28] Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. Controllable invariance through adversarial feature learning. In *Advances in Neural Information Processing Systems*, 2017. 1, 2, 6, 7

[29] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, 2013. 3, 6

[30] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *AAAI/ACM Conference on AI, Ethics, and Society*, 2018. 2