ABSTRACT In this paper, we consider the problem of learning fair data representations that can be used for some downstream utility task in the vendor-user setting. We propose splitting the latent space between sensitive and non-sensitive latent variables where maximum mean discrepancy (MMD) is used to induce statistical independence between sensitive and non-sensitive latent variables. The non-sensitive latent representations can then be used for utility task by the user and achieve group and sub-group fairness with respect to multiple sensitive attributes. We perform extensive experiments and compare the proposed method against various representation learning methods proposed earlier in the literature. Our quantitative results and visualizations show that the proposed method produces representations that are able to achieve better or comparable performance at the utility task while simultaneously achieving sub-group and group fairness.

INDEX TERMS Machine learning, representation learning, fairness, transparency.

I. INTRODUCTION

Machine learning algorithms are increasingly being used in a variety of real-world applications. While these algorithms perform well at complex tasks, they are also capable of exhibiting discriminatory behavior against certain groups in critical fields such as medicine, finance and law [1]. These potentially unethical or illegal outcomes incentivize researchers to study the machine learning models and work towards developing more robust and equitable models. A promising approach towards solving this problem is fair representation learning. We adopt a vendor-user approach [2] where the vendor is responsible for giving fair data representations to the user. The fair representations can be used by the user to perform some downstream utility task. Therefore, the vendor can be optimized to yield fair representations based on some fairness objective and user can be independently optimized to maximize the performance at the utility task.

In this work, we study the problem of learning representations of data that can be used for downstream utility tasks. To achieve this, we use the variational inference setting where the latent space is divided into two subspaces: sensitive and non-sensitive latents. To optimize for fairness objective, we induce statistical independence between sensitive and non-sensitive latents by utilizing the maximum mean discrepancy (MMD). This way the information about sensitive attributes is limited to sensitive latents only and the non-sensitive latents are subsequently used for downstream utility tasks by the user in our vendor-user approach. We perform extensive empirical study and compare the proposed approach to various representation learning methods proposed earlier in the literature. We conduct extensive experiments on real-world image. Our results show that fair data representations learned using the proposed method matches or exceeds the fairness-accuracy trade-off for groups as well as subgroups against the studied baseline works.

II. PRELIMINARIES

A. FAIRNESS CLASSIFICATION

In fair classification, we consider labeled data examples of the form $x, a, t \sim p_{data}$ where $x \in \mathcal{X}$ is the data sample, $a \in \mathcal{A}$ are the sensitive attributes and $t \in \mathcal{Y}$ is the data label that we wish to predict in the utility task. Our goal is to learn a classifier that is predictive of $t$ and satisfies some fairness criteria with respect to the sensitive attribute $a$, i.e. we want predictions that are accurate but not biased in favor of one group or the other. Many fairness criteria have been proposed in the literature over the years and are usually written in terms...
of independence conditions of the various random variables involved. In this work, we focus on demographic parity which states that a classifier is fair when its outcome $\hat{t}$ is independent of the sensitive attribute $a$, i.e. $p(\hat{t} = 1|a = 1) = p(\hat{t} = 1|a = 0)$.

**B. REPRESENTATION LEARNING**

Representation learning is widely used to solve a myriad of complex tasks. The performance of deep learning algorithms generally depend on the quality of representations learned by the model. In this work, our goal is to learn data representations $z$ that achieve fairness with respect to sensitive attributes $a$, i.e. $z \perp a$. Subsequently, any predictor that takes representations $z$ as input, $\hat{t} = g(z)$, also satisfies the demographic parity, $\hat{t} \perp a$.

**C. VARIATIONAL AUTOENCODER (VAE)**

The vanilla variational autoencoder (VAE) [3] is a self-supervised generative model that can be used to learn useful representations from the data. VAE is usually implemented using a Gaussian prior $p(z) = \mathcal{N}(0, I)$ and the learning is performed by maximizing the Evidence Lower Bound (ELBO) which is the lower bound of data log likelihood $\log(p(x))$ for any choice of approximate posterior distribution $q$. ELBO is given as

$$\text{ELBO}(p, q) = \mathbb{E}_{q(z|x)}[\log(p(x|z))] - D_{KL}[q(z|x)||p(z)].$$

Here, $D_{KL}[p||q]$ gives the Kullback–Leibler divergence between two probability distributions $p$ and $q$. The encoder and decoder are also usually implemented as Gaussians, i.e. $q(z|x) = \mathcal{N}(z|\mu_q(x), \Sigma_q(x))$ and $p(x|z) = \mathcal{N}(z|\mu_p(z), \Sigma_p(z))$, respectively. Both encoder and decoder are implemented as neural networks and are learned in an end-to-end manner. However, for modelling binary valued pixels, a Bernoulli decoder can be used i.e. $p(x|z) = \text{Bernoulli}(x|\theta(z))$ with parameters $\theta$.

**D. MAXIMUM MEAN DISCREPANCY (MMD)**

Let’s assume we have two distributions $p$ and $q$ over $X$. Consider a feature map $\phi : X \rightarrow \mathcal{H}$, where $\mathcal{H}$ is called Reproducing Kernel Hilbert Space (RKHS).

If $\mathcal{H}$ satisfies the necessary conditions, we can use kernel trick to compute the inner product in $\mathcal{H}$. For $Y, Z \in X$, we have

$$k(Y, Z) = \langle \phi(Y), \phi(Z) \rangle_\mathcal{H}.$$  

Let $\mu_p(\phi(X)) = \mathbb{E}_{X \sim p}[\phi(X)]$ be the mean embeddings where $X \sim p$. The MMD is given as

$$\text{MMD}(p, q) = \|\mu_p(\phi(X)) - \mu_q(\phi(Y))\|^2_\mathcal{H}.$$  

Since $\|x\|^2 = \langle x, x \rangle$, we have

$$\text{MMD}(p, q) = \langle \mu_p(\phi(X)), \mu_p(\phi(X)) \rangle + \langle \mu_q(\phi(Y)), \mu_q(\phi(Y)) \rangle - 2\langle \mu_p(\phi(X)), \mu_q(\phi(Y)) \rangle.$$  

In the form of kernel function, we can write

$$\text{MMD}(p, q) = \mathbb{E}_p[k(X, X)] + \mathbb{E}_q[k(Y, Y)] - 2\mathbb{E}_{p,q}[k(X, Y)].$$  

This means that we can compute MMD between distributions $p$ and $q$ merely with access to samples $X \sim p$ and $Y \sim q$. For the empirical calculation of MMD given in (2), for $x_i \in X\forall i = 1, \ldots, m$ and $y_i \in Y\forall i = 1, \ldots, m$, we use the following formula:

$$\text{MMD}(X, Y) = \frac{1}{m(m-1)} \sum_{i,j} k(x_i, x_j) - 2\frac{1}{mm} \sum_{i,j} k(x_i, y_j) + \frac{1}{m(m-1)} \sum_{i,j} k(y_i, y_j).$$  

Gaussian kernels have the property that $\text{MMD}(p, q) = 0$ iff $p = q$. In our experiments, we use a Gaussian kernel called Radial Basis Function (RBF). For $x_i, x_j \in X$, RBF kernel value $k(x_i, x_j)$ is given by

$$k(x_i, x_j) = \exp\left(\frac{\|x_i - x_j\|^2}{2\sigma^2}\right).$$

**III. RELATED WORK**

The interest in the field of fairness in machine learning has been increasing in the recent years. The work concerning fairness in machine learning can be broadly classified into group fairness and individual fairness. Reference [2] did the definitive work for individual fairness in vendor-user setting similar to what we use. Reference [4] used a regularized objective to encourage both group and individual fairness. A recent body of work tried to study the dichotomy of individual-group fairness by studying fairness at the intersection of multiple groups and among small individual groups [5, 6].

Some researchers have also explored the problem of fair machine learning by pre- and post-processing of the training dataset [7], [8]. An interesting framework was presented by [9] where the authors propose the data producer, user and regulator with distinct objectives and discuss the properties of fair representations. Generative models, such as variational autoencoders (VAE), were used by [10] to achieve fairness by removing disparities between two sensitive groups using maximum mean discrepancy [11].

However, most body of literature dealing with fairness in machine learning concerns with fairness with respect to a single (often binary) sensitive attribute. Reference [5], [6], and [12] focus on fairness with respect to multiple sensitive attributes. Reference [5] and [6] develop fair classification algorithms based on the notion of identifiable class of subgroups. The main difference is the underlying metric: [5] uses statistical parity while [6] uses calibration. Based on the work of [6], [13] proposed a new algorithm to achieve multi-group fairness by utilizing post-processing boosting.
procedure. Reference [12], on the other hand, took a different approach and proposed a framework that enables flexibly choosing the sensitive attribute of interest at the test time. The authors focus on demographic parity notion of group fairness and introduce flexibly fair representation learning by disentanglement that disentangles information from multiple sensitive attributes. Their flexible and fair variational autoencoder is not only flexible with respect to downstream task labels, but also flexible with respect to sensitive attributes.

Learning fair data representations that are useful for utility task and simultaneously avoid the harmful interference of sensitive attributes has been a popular topic of research in the fairness community. Reference [10] treat the sensitive variable as a nuisance variable and try to produce fair representations by removing the information about sensitive variables from the representations. In [14] authors also build upon VAE and propose a VAE architecture called DB-VAE, which learns sensitive latent variables that can bias the model (e.g., skin tone, gender) and propose an algorithm on top of this DB-VAE using these latent variables to debias systems like facial detection systems. In [15], the authors propose a method called LAFTR that uses an adversary to learn fair representations. Reference [12] propose learning fair representations by disentangling the sensitive information in the latent space from the non-sensitive information. The authors in [16] propose algorithm that aims to simultaneously ensure accuracy parity and equalized odds. The main idea underlying the design of their algorithm is to align the conditional distributions of representations (rather than marginal distributions) and use the balanced error rate (i.e., the conditional error) on both the target variable and the sensitive attribute.

Our proposed method can be classified under the fair representation learning. Similar to FFVAE [12], our representations are flexible, i.e. the user can decide which attribute is to be considered as sensitive at the test time. However, FFVAE [12] uses disentanglement in the latent space to produce fair representations while we induce statistical independence between sensitive and non-sensitive latents in the latent space. LAFTR [15] uses adversarial training to produce fair representations which often causes stability issues [17]. Unlike LAFTR [15], our method does not use adversarial training. Reference [10] used maximum mean discrepancy to “stamp out” the information about sensitive attribute from the latent representation. Our work differs fundamentally from [10] because we split the latent space into sensitive and non-sensitive latent and induce statistical independence between them.

IV. METHOD

In this section, we present the details of our proposed method. The proposed method is conceptually presented in Figure 1. We use variational autoencoder (VAE) which is used to learn data representations in a self-supervised manner by trying to reconstruct the input $x$ from the learned latent representation $[z, b]$. The latent space is divided into two parts, the non-sensitive latent variable $z \in \mathbb{R}^s$ and sensitive latent variable $b \in \mathbb{R}^t$. Both encoder and decoder are parameterized as neural networks with parameters $\phi$ and $\theta$, respectively. The ELBO that we maximize that learn the parameters $\phi$ and $\theta$ is given as,

$$
\text{ELBO}(\phi, \theta) = \mathbb{E}_q[\log p_\theta(x|z, b)] - D_{KL}[q_\phi(z, b|x)||p(z, b)].
$$

The encoder outputs the parameters of the joint distribution of variables $z$ and $b$, i.e. $q(z, b)$. We want to ensure that the non-sensitive latent variable $z$ is independent of any information related to the sensitive attributes $a$ while useful enough to be used for downstream utility task. To achieve this, we induce the statistical independence between the random variables $z$ and $b$ by minimizing the maximum mean discrepancy (MMD) between the joint distribution $q(z, b)$ and the product of their marginals, i.e. $q(z)\prod_j q(b_j)$. Note that we use the product of marginals to represent the distribution of sensitive latent $b$, i.e.

$$
q(b) = \prod_j q(b_j).
$$

Using $q(b)$ as given in (5) ensures that we can chose which sensitive attributes we care about at the test time. For example, we can achieve fairness with respect to a subset of sensitive attributes $a_i \land a_j \land a_k$ simply by removing the corresponding sensitive latents from the latent representation and using the representation $[z, b] \setminus \{b_i, b_j, b_k\}$ for downstream utility task.

The non-sensitive or fair latent representation $z$ for corresponding input $x$ is used to predict the target variable $t$. This is achieved by maximizing the probability $p_t(t|z)$ parameterized by parameters $\zeta$. More details about the implementation are presented in Section V.

To make the sensitive latent $b$ informative about sensitive attributes $a$, we maximize the likelihood of $a$ given $b$, i.e. $p_a(a|b)$ with parameters $\lambda$.

The final objective that we maximize is given as

$$
L_{\text{final}}(\phi, \theta, \lambda, \zeta) = \text{ELBO}(\phi, \theta)
- \text{MMD}(q(z, b)||q(z)\prod_j q(b_j))
+ p_t(t|z)
+ p_a(a|b).
$$

V. IMPLEMENTATION

In this section, we discuss the implementation details of our proposed method to learn fair data representations. The representations are learned using a variational autoencoder (VAE) [3], as shown in Figure 1, where we use convolutional neural networks to implement both the encoder and the decoder. Similar to [12], we assume a variational posterior that factorizes across $z$ and $b$, i.e.

$$
q(z, b|x) = q(z|x)q(b|x)
$$

The encoder outputs the parameters for the variational posterior distribution for some input $x$, i.e.
\[ \mu_q(x), \Sigma_q(x), \Lambda_q(x) \] = Encoder_q(x). The posterior factors in equation (7) are then given as \( q(z|x) = \mathcal{N}(z|\mu_q(x), \Sigma_q(x)) \) and \( q(b|x) = \mathcal{N}(b|\Lambda_q(x), I) \).

To realize the term MMD\( (q(z, b)||q(z) \prod_{j} q(b_j)) \) in equation (6), we need samples from distributions \( q(z, b) \) and \( q(z) \prod_{j} q(b_j) \). Samples \( [z, b] \sim q(z, b) \) can be easily obtained from the aggregate posterior. To obtain the samples \( [z', b'] \sim q(z) \prod_{j} q(b_j) \), we use the method used in [12]. For i-example, the latent code \( [z', b'] \) is split along the dimensions of \( b \) as \( [z', b'_1, b'_2, \ldots, b'_i] \), the minibatch index order for each subspace is then randomized, simulating samples from the product of marginals. Having access to samples \( [z, b] \sim q(z, b) \) and \( [z', b'] \sim q(z) \prod_{j} q(b_j) \), we can use equation (3) to evaluate the MMD value.

To perform the utility task, we maximize the probability \( p_{\xi}(t|z) \). The probability \( p_{\xi}(t|z) \) is modelled by the utility network, with parameters \( \xi \), which is in fact an MLP in our experiments, i.e.

\[ \hat{t} = \text{UtilityNetwork}(z; \xi). \]

The parameters of the utility network, \( \xi \), are learned in an end-to-end fashion by optimizing the utility loss function \( L^c_{\text{utility}}(t, \hat{t}) \) where \( L_{\text{utility}} \) can be any appropriate loss function depending on the nature of the target variable \( t \). For example, we can use the cross entropy loss as the utility loss function if the target \( t \) is binary.

We use a factorized prior for latent variables. The prior is given as \( p(z, b) = p(z)p(b) \). In our implementation, similar to [12], we use standard Gaussian distribution for prior \( p(z) \) and Uniform distribution for \( p(b) \).

We maximize the probability \( p_{\xi}(a|b) \) to make the sensitive latent \( b \) informative about sensitive attribute \( a \), see Section IV for details. The implementation is based on an MLP which is used to predict the sensitive attributes \( a \) while taking sensitive latent \( b \) as the input, i.e.

\[ \hat{a} = \text{MLP}(b; \lambda). \]

The parameters \( \lambda \) for MLP are learned in an end-to-end fashion by minimizing the loss function \( L^b_{\text{sensitive}}(a, \hat{a}) \). \( L_{\text{sensitive}} \) depends on the nature of \( a \), for example we can use cross entropy loss for binary sensitive attributes and mean squared error (MSE) loss to regress a continuous sensitive attribute (for example age). If there are multiple sensitive attributes, say \( m \), then we model \( b \in \mathbb{R}^s \) such that \( s = m \) and \( b \) where \( i = 1, 2, \ldots, m \) corresponds to \( a_i \) where \( i = 1, 2, \ldots, m \). To predict each \( a_i \), we use a separate MLP such that \( \hat{a}_i = \text{MLP}(b_i) \) where \( i = 1, 2, \ldots, m \). This way we can ensure that the user can decide which sensitive attribute is of concern at the test time and can use the representation \( [z, b] \setminus \{b_i\} \) for the utility task. We present the development for the case of a single sensitive attribute, \( m = 1 \), but the case for multiple sensitive attributes is straightforward to generalize.

The final objective, given in equation (6), can now be given in terms of loss functions as,

\[ L_{\text{final}}(\phi, \theta, \lambda, \xi) = \text{ELBO}(\phi, \theta) \]

\[ -\text{MMD}(q(z, b)||q(z) \prod_{j} q(b_j)) \]

\[ -L^c_{\text{utility}}(t, \hat{t}) \]

\[ -L^b_{\text{sensitive}}(a, \hat{a}), \]
where $\text{ELBO}(\phi, \theta)$ is given in equation (4). We use ADAM optimization algorithm [18] to maximize the objective given in equation (8).

VI. EXPERIMENTS
In this section, we provide the details about our experimental settings and the experimental results.

A. TRAINING DATA
The goal of this study is to learn fair data representations. We perform extensive experiments on a dataset consisting of $4 \times 10^5$ images. Similar to [14], we generate our dataset with $2 \times 10^5$ positive images (images that contain human face) and an equal number of negative images (images of non-faces). We split the dataset into 80% and 20% ratio into training and validation sets, respectively. Positive sample images are taken from CelebFaces Attributes Dataset (CelebA)\(^1\) dataset and cropped to a square based on the annotated face bounding box. Negative sample images are taken from a wide variety of non-human categories from ImageNet dataset [19]. All images are resized to $64 \times 64$ resolution. All CelebA images are assigned target label $t = 1$ because they contain human face while all images taken from ImageNet dataset are assigned target label $t = 0$. The utility task is to predict if an image contains a human face or not. The sensitive attributes considered include sex (male or female) and skin tone (lighter or darker).

B. TEST DATA
After performing the training on the training dataset generated as explained in Section VI-A, we test the performance of the trained models on PPB test dataset [20], PPB dataset consists of images of 1270 male and female parliamentarians from various African and European countries. Images are consistent in pose, illumination, and facial expression, and the dataset exhibits parity in both skin tone and gender. The gender of each face is annotated with either “Male” or “Female” label and the skin tone annotations are based on the Fitzpatrick skin type classification system [21] with each image labeled as either “Lighter” or “Darker”.

We compare the proposed method against various baselines. The studied baselines include FFVAE [12], LAFFTR [15] and CLFR [16].

C. TRAINING
The models are trained to predict if the given image contains a human face or not. The sensitive attributes considered include sex (male or female) and skin tone (lighter or darker). Since the target variable we wish to predict is binary, we use binary cross entropy loss to train the utility network, i.e.

$$L_{utility}^c = -(t \log(\hat{t}) + (1 - t) \log(1 - \hat{t})),$$

where $t$ is the ground truth label and $\hat{t}$ is the predicted probability of $t$.

\(^1\)http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

Likewise, to learn the parameters $\lambda$ of the MLP network which is used to predict the sensitive attributes, see Figure 1 and Section V, we use cross entropy loss, i.e.

$$L_{\text{sensitive}}^\lambda = -a^T \log(\hat{a}),$$

where $a \in \mathbb{R}^{1 | A}$ is the vector encoding the ground truth labels for $|A|$ number of sensitive attributes and $\hat{a}$ contains their predicted probabilities.

The final objective given in equation (8) is maximized to learn the representations.

D. EVALUATION METRIC
The fundamental evaluation metric that we use is the accuracy of the predicted target. The higher the accuracy, the better the model performance. Since there are multiple groups for each experiment, we also take the mean and the variance of the accuracies across all groups. The higher the mean accuracy and the lower the variance, the better the model performance.

E. QUANTITATIVE RESULTS
We have two sensitive attributes, gender and skin tone. We trained models to predict the target variable $t$ (whether an image contains human face). At the test time, we test the trained models under three different settings depending on which sensitive attribute is considered. i.e.

• skin tone: only skin tone is considered as the sensitive attribute,
• gender: only gender is considered as the sensitive attribute,
• skin tone and gender: both skin tone and gender are considered as sensitive attributes.

First we consider only the skin tone as the sensitive attribute and test the performance of the trained models on PPB test dataset. For this experiment, we use $[\hat{a}, \hat{b}] \setminus \{b_{\text{skin tone}}\}$ as our representations. The results of this experiment are shown in Table 1. We can see that the proposed

| Methods | Accuracy (%) | Mean (%) | Variance |
|---------|--------------|----------|----------|
| Ours    | 96.2         | 87.9     | 92.1     | 17.6     |
| FFVAE [12] | 97.5       | 85.1     | 91.3     | 38.44    |
| LAFFTR [15] | 95.5       | 84.3     | 89.9     | 31.36    |
| CLFR [16] | 93.5       | 86.6     | 88.6     | 24.5     |

Likewise, to learn the parameters $\lambda$ of the MLP network which is used to predict the sensitive attributes, see Figure 1 and Section V, we use cross entropy loss, i.e.

$$L_{\text{sensitive}}^\lambda = -a^T \log(\hat{a}),$$

where $a \in \mathbb{R}^{1 | A}$ is the vector encoding the ground truth labels for $|A|$ number of sensitive attributes and $\hat{a}$ contains their predicted probabilities.

The final objective given in equation (8) is maximized to learn the representations.

D. EVALUATION METRIC
The fundamental evaluation metric that we use is the accuracy of the predicted target. The higher the accuracy, the better the model performance. Since there are multiple groups for each experiment, we also take the mean and the variance of the accuracies across all groups. The higher the mean accuracy and the lower the variance, the better the model performance.

E. QUANTITATIVE RESULTS
We have two sensitive attributes, gender and skin tone. We trained models to predict the target variable $t$ (whether an image contains human face). At the test time, we test the trained models under three different settings depending on which sensitive attribute is considered. i.e.

• skin tone: only skin tone is considered as the sensitive attribute,
• gender: only gender is considered as the sensitive attribute,
• skin tone and gender: both skin tone and gender are considered as sensitive attributes.

First we consider only the skin tone as the sensitive attribute and test the performance of the trained models on PPB test dataset. For this experiment, we use $[\hat{a}, \hat{b}] \setminus \{b_{\text{skin tone}}\}$ as our representations. The results of this experiment are shown in Table 1. We can see that the proposed
method performs comparable to FFVAE [12] while performing better than other two baselines for “Lighter” faced images. For “Darker” face images, the proposed method performs better than all the baselines. The mean accuracy is also higher for our method and it also exhibits the lowest variance.

For the second experiment, we only consider gender as the sensitive attribute. The representations that we use are \([z, b] \setminus \{b_{\text{gender}}\}\). The results of this experiment are given in Table 2. Our method perform better than all the baselines for both “Male” and “Female” faces. The proposed method also shows higher mean accuracy and lower variance as compared to the baseline methods.

In the third experiment, we consider both the skin tone and the gender as the sensitive attributes. We now have four groups: dark male, dark female, light male and light female. The results for this experiment are given in Table 3. We can see that the proposed method achieves better accuracy for...
TABLE 3. Result for experiment when both skin tone and gender are considered as sensitive attributes. We have four groups: Dark Male, Dark Female, Light Male and Light Female.

| Methods       | Dark Male | Dark Female | Light Male | Light Female | Mean (%) | Variance |
|---------------|-----------|-------------|------------|--------------|----------|----------|
| Ours          | 90.3      | 87.5        | 98.8       | 96.3         | 93.2     | 20.5     |
| FFVAE [12]    | 86.3      | 83.9        | 97.9       | 97.1         | 91.3     | 39.2     |
| LAFTR [15]    | 85.1      | 81.5        | 94.7       | 90.4         | 87.92    | 25.3     |
| CLFR [16]     | 85.2      | 83.8        | 95.6       | 94.2         | 89.53    | 27.5     |

three of the four groups while performing comparable to FFVAE [12] on Light Female group. Overall, our method achieves the best mean accuracy for all population groups and the lowest variance as compared to the studied baselines.

F. VISUALIZATION OF LEARNED REPRESENTATIONS

To visualize the learned representations, we use t-SNE [22] algorithm to visualize the quality of the learned representations in the latent space. We visualize the case where skin tone is taken as the sensitive attribute. The entire data is divided into two groups, light faces and dark faces. We perform the following steps to visualize the test data representations.

- first, each test image is passed through the variational autoencoder to get the corresponding latent space representation. For visualization, we use \([z, b] \setminus \{b_{\text{skin tone}}\}\) as our representation vector.
- second, the latent space representation vector is embedded into 2D space using t-SNE algorithm.
- third, all 2D vectors corresponding to all images are visualized by plotting.

We take the centroids of both groups, light faces and dark faces, in 2D visualization space and find the distance between them. Since good latent space representations are devoid of information about sensitive attribute, the distance between the centroids is smaller for better representations. Let \(c_{\text{dark}}\) be the centroid of dark faces group and \(c_{\text{light}}\) be the centroid of light faces group. We define \(\Delta c = |c_{\text{dark}} - c_{\text{light}}|\) as distance between the two centroids.

The visualizations are shown in Figure 2 for our proposed method, FFVAE [12], (c) LAFTR [15] and (d) CLFR [16] for light faces and dark faces groups. We visualize 150 randomly sampled representations for each group. Figure 2 also shows \(\Delta c\) values for our proposed method and the baselines. We see that \(\Delta c\) value for our proposed representation learning method is the lowest among all the studied methods. This shows that our representations have the least amount of sensitive information embedded in them while the results presented in Section VI-E show that our representations perform better or comparable than the baselines at the utility task as well.

VII. DISCUSSION

In this work, we have proposed a method to learn fair data representations that can be used for downstream utility tasks. We have compared our method against three baselines: FFVAE [12], LAFTR [15] and CLFR [16]. We performed extensive experiments on a custom dataset [14] where the utility task was to predict whether the given image contains human face or not. Similar to FFVAE [12], our representations are also flexible such that the user can decide at the test time which attribute is to considered as sensitive. We performed experiments under three different test settings with different sensitive attributes at the test time: skin tone only, gender only and skin tone and gender both. Note that the same representations were used for all aforementioned test settings. Our results showed that the proposed method performs better or comparable to the baselines in all three settings.

We also visualized the learned representations using t-SNE [22] algorithm. We showed in Section VI-F that the our method produces representations that have the least content of sensitive information embedded into them as compared to the baseline methods. However, the experiment results presented in Section VI-E show that our representations are informative about the utility task as they perform better or comparable than the baseline methods at performing the utility task. This shows that our method produces representations that retain the useful content about the utility task while minimizing the information about sensitive attributes.

All three baseline methods, FFVAE [12], LAFTR [15] and CLFR [16], use adversarial learning in their training phase. Adversarial training is difficult and suffers from stability issues [17]. This makes training the networks difficult and it also affects the performance of the resultant representations. Our method, however, does not use any adversarial objective in the training which makes the training process simpler.

VIII. CONCLUSION

In this work, we presented a method to learn fair data representations that can be used for down stream utility tasks, such as classification. The learned data representations are flexible with respect to the sensitive attributes, i.e. it can be chosen at the test time which attributes are to be treated as sensitive attributes. We have performed extensive experiments to compare the proposed method against other fair representation learning methods in literature. Our experimental results show that the representations learned using the proposed method perform better or comparable to other studied methods on classification tasks. Note that we use the same representations for different sets of sensitive attributes for the classification experiments. This shows that our representations are flexible and can be used to achieve fairness objectives for different sensitive attributes at the test time. We also visualize the
representations in the embedding space and show that our representations achieves greater independence with respect to the sensitive attributes.

REFERENCES

[1] S.-E. Kim, H. Y. Paik, H. Yoon, J. E. Lee, N. Kim, and M.-K. Sung, “Sex- and gender-specific disparities in colorectal cancer risk,” World J. Gastroenterol., vol. 21, pp. 75–5167, 2015.

[2] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, “Fairness through awareness,” 2011, arXiv:1104.3913.

[3] D. P. Kingma and M. Welling, “Auto-encoding variational Bayes,” 2014, arXiv:1312.6114.

[4] R. S. Zemel, L. Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, “Learning fair representations,” in Proc. 30th Int. Conf. Mach. Learn., 2013, pp. 325–333.

[5] M. Kearns, S. Noel, A. Roth, and Z. S. Wu, “Preventing fairness gerrymandering: Auditing and learning for subgroup fairness,” in Proc. 35th Int. Conf. Mach. Learn., 2018, pp. 2564–2572.

[6] U. Hébert-Johnson, M. P. Kim, O. Reingold, and G. N. Rothblum, “Calibration for the (computationally-identifiable) masses,” 2017, arXiv:1711.08513.

[7] F. D. P. Calmon, D. Wei, B. Vinzamuri, K. N. Ramamurthy, and K. R. Varshney, “Optimized pre-processing for discrimination prevention,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 1–10.

[8] S. Hajian, J. Domingo-Ferrer, A. Monreale, D. Pedreschi, and F. Giannotti, “Discrimination- and privacy-aware patterns,” Data Mining Knowl. Discovery, vol. 29, pp. 1733–1782, Nov. 2015.

[9] D. McNamara, C. S. Ong, and R. C. Williamson, “Provably fair representations,” 2017, arXiv:1710.04394.

[10] C. Louizos, K. Swersky, Y. Li, M. Welling, and R. S. Zemel, “The variational fair autoencoder,” 2016, arXiv:1511.00830.

[11] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, “A kernel method for the two-sample-problem,” in Proc. Adv. Neural Inf. Process. Syst., 2006, pp. 1–8.

[12] E. Creager, D. Madras, J.-H. Jacobsen, M. A. Weis, K. Swersky, T. Pitassi, and R. Zemel, “Flexible fair representation learning by disentanglement,” 2019, arXiv:1906.02589.

[13] M. P. Kim, A. Ghorbani, and J. Y. Zou, “Multiaccuracy: Black-box post-processing for fairness in classification,” in Proc. 2019 AAAI/ACM Conf. AI, Ethics Soc., 2019, pp. 247–254.

[14] A. Amini, A. P. Soleimany, W. Schwarting, S. N. Bhatia, and D. Rus, “Uncovering and mitigating algorithmic bias through learned latent structure,” in Proc. AAAI/ACM Conf. AI, Ethics Soc., 2019, pp. 289–295.

[15] D. Madras, E. Creager, T. Pitassi, and R. S. Zemel, “Learning adversarially fair and transferable representations,” in Proc. 35th Int. Conf. Mach. Learn., 2018, pp. 3384–3393.

[16] H. Zhao, A. Coston, T. Adel, and G. J. Gordon, “Conditional learning of fair representations,” 2019, arXiv:1910.07162.

[17] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training GANs,” 2016, arXiv:1606.03498.

[18] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2015, arXiv:1412.6980.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.

[20] J. Buolamwini and T. Gebru, “Gender shades: Intersectional accuracy disparities in commercial gender classification,” in Proc. 1st Conf. Fairness, Accountability Transparency, 2018, pp. 77–91.

[21] T. B. Fitzpatrick, “The validity and practicality of sun-reactive skin types I through VI,” Arch. Dermatol., vol. 124, no. 6, pp. 869–871, 1988.

[22] L. van der Maaten and G. Hinton, “Visualizing data using t-SNE,” J. Mach. Learn. Res., vol. 9, pp. 2579–2605, Nov. 2008.

MUHAMMAD USAMA received the B.S. degree in electrical engineering from the National University of Sciences and Technology (NUST), in 2017, and the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), in 2019. He is currently pursuing the Ph.D. degree with the School of Electrical Engineering, KAIST. His current research interests include data driven control, machine learning, reinforcement learning, deep learning, and fairness and transparency in machine learning.

DONG EUI CHANG (Member, IEEE) received the B.S. degree in control and instrumentation engineering and the M.S. degree in electrical engineering from Seoul National University and the Ph.D. degree from Caltech. He was an Associate Professor of applied mathematics at the University of Waterloo. He joined at KAIST, in 2017, where he is currently a Full Professor of electrical engineering with KAIST. He has coauthored of the deep learning book Deep Neural Networks in a Mathematical Framework (Springer, 2018). His research interests include control, mechanics, and AI.

***