Learning Fair and Invariant Representation with Hilbert Schmidt Independence Criterion

Prince Zizhuang Wang
University of California Santa Barbara
zizhuang_wang@ucsb.edu

Yu-Xiang Wang
University of California Santa Barbara
yuxiangw@cs.ucsb.edu

William Yang Wang
University of California Santa Barbara
william@cs.ucsb.edu

Abstract

In this paper, we study the problem of learning fair representation that is invariant to certain sensitive factors. This problem has drawn a lot of attention recently as learning invariant representation is useful for reducing bias in many machine learning problems. Unfortunately, recent adversarial approach often comes with the trade-off between fairness and model performance. Instead of following previous adversarial approaches, we propose to learn fair representation with Hilbert Schmidt Independence Criterion with the help of kernel methods. We prove that the resulting model parameterized by neural network indeed learns the appropriate kernel that can be used for fairness measurement. We argue that the learned representation is in fact invariant to unwanted variations. With the results on previous benchmarks for learning fair and invariant representation, we demonstrate that our method eliminates noisy factors while achieving state-of-the-art and competitive results on various benchmarks.

1 Introduction

Learning informative representations that eliminates nuisance factors of variation is one of the most important problems in machine learning. In computer vision, many algorithms seek to learn features that are invariant to certain factors such as scaling and shifting, which can help tasks like object classification and detection. The same problem is also faced by many decision-making models, in which fairness is often as important as model performance. Fair machine learning [31, 29, 23, 21] has drawn a lot of attention only until recently. The two important goals of learning fair representations can be categorized as achieving group fairness and individual fairness [31]. Group fairness has been studied by [12] before. It requires the proportion of each group receiving positive classification is the same to the proportion of the entire population. This criterion may be unfair to certain individuals in some groups, where the algorithms may favor individuals in protected groups while ignoring individuals who should be receiving positive outcome. The other criterion, achieving individual fairness, addresses this issue by seeking individuals who will be classified positive for certain tasks regardless of which group they come from. This is largely related to the problem of learning invariant representations, as we wish the classification to be invariant to unwanted or sensitive traits of the targets.

In recent years, the problem of learning fair and invariant representations has been largely studied with the help of deep neural networks [19, 16]. These deep architectures can easily uncover many latent features of the data. Convolutional neural network (CNN) [18] is one particular deep neural network that is designed to learn features that eliminates noisy factors. Its weight sharing property

Preprint. Under review.

and pooling operation [16] ensure that those high-level features are invariant to scaling and shifting of images, which helps many machine learning algorithms to gain a lot of success in computer vision [16]. Nevertheless, these deep learning strategies are still feature-engineering techniques which often lack interpretability and explainability. This would be a problem if one wants to develop more advanced algorithms with the help of backbone theories.

Alternatively, one can construct probabilistic models to learn a representation z that is statistically independent of noisy random variable s. Zemel et al. [31] seeks to learn representations that satisfy statistical parity, which is mainly for group fairness and could be unfair to certain individuals. Louizos et al. [21] proposes Variational fair Autoencoder (VFAE), a variant of Variational Autoencoder (VAE) [15], to provide fair encoding that eliminates information of sensitive factors and can be further used to achieve fairness in supervised classification. Xie et al. [29] and Kamnitsas et al. [13] make use of adversarial training [6] to factor out unwanted factors from the learned representations. The objective is a game theory between two classification models based on the learned representations, one of which is associated with the main task, and the other with the unwanted factor s. The two models will compete with each other. In the final equilibrium state, it is expected that the second model will fail to predict nuisance factors with good accuracy, in which case the learned representations contain no information of nuisance factors. Although adversarial training approaches are promising and achieved a lot of success as a probabilistic models [6] in recent years, its training objective often poses instability during training. Moyer et al. [23] proposes to eliminate protected factors from features by minimizing the mutual information between protected factors and the learned representations directly. The resulting simple objective is more stable to train.

In this paper, we propose a simple, elegant framework to learn fair and invariant features that is easy to train. We make use of kernel methods and Hilbert Schmidt Independence Criterion (HSIC) [8, 10]. HSIC is a powerful statistical tool for measuring independence between random variables. Our major contributions can be categorized as the following:

1. We propose a *min-max* Hilbert Schmidt Independence Criterion game with the help of kernel method to learn fair and invariant representations.
2. We demonstrate the simplicity and elegance of our model objective. We prove that the algorithm does find an appropriate kernel for HSIC. Compared with adversarial training approaches and previous kernel methods [24], the trade-off between fairness and model performance is alleviated.
3. We show the strength of our model empirically. Based on previous benchmarks on fair classification problem and learning invariant representations, we show that our model achieves competitive and state-of-the-art results on various metrics compared with previous fair machine learning models.

2 Hilbert Schmidt Independence Criterion

We will have a short review on some of the concepts of Reproducing Kernel Hilbert Space (RKHS), cross-covariance operators, and Hilbert Schmidt norm. We also review empirical evaluation for Hilbert Schmidt Independence Criterion.

2.1 Reproducing Kernel Hilbert Space

A Hilbert space (feature space) H of functions is an inner product space that is complete with respect to the distance metric defined by the associated inner product. A kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a function associated with a Hilbert space such that

$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle_H, \quad x, x' \in \mathbb{R}$$

where $\langle \cdot, \cdot \rangle_H$ is an inner product defined on H, and $\Phi(x) : \mathcal{X} \to H$ is a feature map that projects $\mathcal{X}$ into the Hilbert space H. The kernel function constructed in this way is positive definite.

With the help of kernel function, it is possible to evaluate functions as an inner product in some feature spaces. These feature spaces are called Reproducing Kernel Hilbert Space. A Reproducing Kernel Hilbert Space (RKHS) $\mathcal{F}$ is a Hilbert space with the *reproducing property* [26]

$$\langle \Phi(x), f \rangle = \langle k(\cdot, x), f \rangle = f(x), \quad \forall f \in \mathcal{F}, \forall x \in \mathbb{R}$$

where feature map $\Phi(x) = k(\cdot, x)$ is also a function in $\mathcal{F}$. And the kernel in RKHS $k(x_i, x_j) = \langle k(\cdot, x_i), k(\cdot, x_j) \rangle$ is called the reproducing kernel. This property can be drived from *Riesz representation theorem* [7]. It is easy to see the feature maps span the feature space $\mathcal{F}$. This property is particularly helpful for infinite dimensional feature space $\mathcal{F}$ as we can now express the evaluation at a point $x \in \mathbb{R}$ of any function in $\mathcal{F}$ as a linear combination of feature maps in $\mathcal{F}$, that is,

$$f(x) = \sum_{i=1}^{n} \alpha_i k(x_i, x), \quad x_i, x \in \mathbb{R}$$

What makes RKHS useful in our case is the well-known *Moore–Aronszajn theorem* [1].

Theorem 1 (Moore-Aronszajn)**.** *For any positive definite kernel k, there is a RKHS of functions on $\mathcal{X}$ for which k is its reproducing kernel.*[1]

This theorem is extremely helpful since once we have a kernel function, given that is positive definite, we can always find a feature space in which every member of it can be easily expressed as a summation of terms with respect to the kernel. We shall see the usefulness of RKHS in later sections of this paper.

2.2 Hilbert Schmidt Independence Criterion

Before we introduce Hilbert Schmidt Independence Criterion (HSIC), we first review the definition of *Hilbert Schmidt norm* (HS norm), which allows us to define HSIC. Given an operator $C : \mathcal{G} \to \mathcal{F}$, and othonormal bases $\{f_j\}_j, \{g_i\}_i$ of $\mathcal{F}$ and $\mathcal{G}$ respectively, the Hilbert Schmidt norm of this operator is defined as,

$$||C||_{HS} = \sqrt{\sum_{i,j} \langle Cg_i, f_j \rangle_{\mathcal{F}}^2}$$

An operator is called a Hilbert Schmidt (HS) operator if its HS norm exists. One such operator that serves our interests particularly is the so-called *cross-covariance operator* [8]. Let $\mathcal{F}$ and $\mathcal{G}$ be two Reproducing Kernel Hilbert Spaces with kernel functions k and l respectively, the cross-covaiance operator $C_{xy} : \mathcal{G} \to \mathcal{F}$ is defined as a mapping from one RKHS to the other RKHS such that for $\Phi \in \mathcal{F}, \Psi \in \mathcal{G}$

$$C_{xy} = \mathbb{E}_{xy}[\Phi(x) \otimes \Psi(y)] - \mu_x \otimes \mu_y$$

where $\mu_x = \mathbb{E}_x[\Phi(x)], \mu_y = \mathbb{E}_y[\Psi(y)]$.

The cross-covariance operator can be used to generalize covariance matrix in finite dimensional space to infinite dimensional function space. According to Feuerverger [3], the largest singular value of $||C_{xy}||$ is zero if and only x is statistically independent of y. Hence, the cross-covariance operator can be used to induce criterions for measuring degree of independence between random variables. A more promising criterion is to consider the sum of all squared singular values $||C_{xy}||_{HS}^2$ rather than just taking the largest one. This leads to the definition of *Hilbert Schmidt Independence Criterion* [8].

Definition 1 (Hilbert Schmidt Independence Criterion)**.** *Given two universal RKHS $\mathcal{F}$ and $\mathcal{G}$ with joint distribution p_{xy}, the Hilbert Schmidt Independence Criterion is defined as squared HS norm of the associated cross-covariance operator C_{xy},*

$$HSIC(p_{xy}, \mathcal{F}, \mathcal{G}) = ||C_{xy}||_{HS}^2$$

Now, to see how HSIC relates to dependence among random variables, we first need to give the definition of *universal kernel*.

Definition 2 (Universal kernel)**.** *Given a compact domain $\mathcal{X}$, a kernel function k is universal if the feature map $\Phi(x) = k(\cdot, x)$ is continuous for $\forall x$ and that the RKHS associated with k is dense in $C(\mathcal{X})$, where $C(\mathcal{X})$ is a space of continuous and bounded functions on $\mathcal{X}$.*

Theorem 2 (Independence)**.** *Given two RKHSs $\mathcal{F}, \mathcal{G}$ with universal kernels k, l on $\mathcal{X}, \mathcal{Y}$. If any function in $\mathcal{F}$ and $\mathcal{G}$ is bounded, then $HSIC(p_{xy}, \mathcal{F}, \mathcal{G}) = 0$ if and only if x and y are statistically independent*[2]*.*

[1] Proof can be found at [1]

[2] Proof can be found at [8]

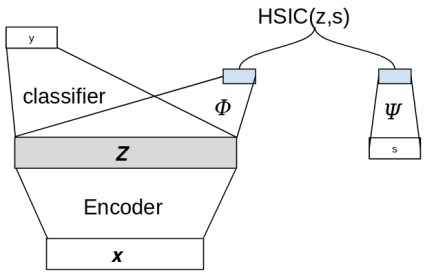


Figure 1: Network Architecture of our model for learning informative representation z with respect to y which factors out protected factor s.

This theorem [8] makes HSIC an ideal tool in our problem of learning invariant representations, as we can include it as a constraint or penalty term during training to restraint z from correlating with the protected factor s. To see the easiness of using HSIC, we now review an efficient way to calculate it.

Lemma 1. *Empirical HSIC Given m sample points, we can compute HSIC empirically in $O(m^2)$ with the following formula*[3]*,*

$$HSIC(\mathcal{F}, \mathcal{G}) = \frac{1}{(m-1)^2} Tr(KHLH)$$

where $K = \Phi\Phi^T$, $L = \Psi\Psi^T$, and $H = I - 11^T/m$

One major drawback of all kernel methods is the computational cost as a kernel has to be evaluated with all pair of points. This is also seen by the $O(m^2)$ running time of empirical HSIC. However, we shall see that with kernel approximation technique and online SGD [2], we can compute HSIC efficiently with batch training, in which m will be the limited batch size which makes $O(m^2)$ negligible. In the next section, we introduce details of our model with thorough analysis.

3 Invariant Representation Learning with HSIC

3.1 Theoretical Analysis

The problem of learning invariant and fair representations can be formulated as the following. Given a set of samples $(x_i, y_i), i = 1, 2, ..., N$, with some protected or unwanted factors of variation s, we want to learn a representation embedding z such that our model $f(z) = y'$ performs well on task of predicting y, while eliminating information of s. In other words, the resulting $y' = f(z)$ is conditionally independent of s given z. In fair machine learning, the protected factors s could be race, gender, or some other sensitive attributes of individuals. Ideally, a fair prediction model should make the same prediction no matter what s might be. Some adversarial approaches [29, 13] have been proposed to deal with this problem.

We deploy the Hilbert Schmidt Independence Criterion (HSIC) that we have introduced in the last section. Here we have three variables, target y, sensitive factor s, and the latent representation z. The model architecture is shown in Figure 2. We parameterize the *encoder*, feature map Φ, and prediction model $f : \mathcal{Z} \to \mathcal{Y}$ as neural networks. The objective is to minimize the classification cost for predicting y with a $HSIC$ penality term,

$$\mathcal{L}(x, y) = -\mathbb{E}_{z \sim enc(x)}[\log f(y|z)] + HSIC_{z \sim enc(x)}(z, s)$$

To make z invariant to s, we need to learn an encoder such that it eliminates information of s while mapping input to z. We achieve this by adding a constraint with respect to HSIC. Since $\Phi(\cdot)$ is a neural network with non-linearity, it maps input to a finite-dimensional space $\mathbb{R}^D$. We thus choose the same strategy as in [25, 17], in which we approximate the kernel function with low-dimensional non-linear mapping as

$$k(x, y) \approx \Phi(x)^T \Phi(y)$$

[3] See [8] for a complete discussion and proof.

induced by Euclidean inner product. This kernel is always positive definite, so there will always be an associated RKHS. To see how we relate this with HSIC and fair or invariant representation learning, we will make the following claim.

Claim 1. *The above Euclidean kernel approximation method with feature map $\Phi(\cdot)$ recognized by neural network with sigmoid or rectified [5] non-linearity, induces a bounded universal kernel and a RKHS associated with it. The resulting kernel, given s and its associated universal RKHS $\mathcal{G}$, can be used with Hilbert-Schmidt Independence Criterion $||C_{zs}||^2_{HS}$ for which $||C_{zs}||^2_{HS}= 0$ if and only if z, s are independent.*

Proof. We give a short but concise proof of the above claim. First, since the feature map $\Phi(\cdot)$ is recognized by neural network with *sigmoid* or *rectified* non-linearity, the function $k(\cdot, \cdot)$ is positive definite and is therefore a kernel function. Since k is positive definite, by Theorem 1, there must exists a RKHS $\mathcal{F}$ associated with k. To see that this kernel is universal according to Definition 2, we need to show that $\Phi(\cdot)$ is continuous everywhere and that RKHS is dense in $C(\mathcal{Z})$, where $C(\mathcal{Z})$ is the space of continuous function on compact set $\mathcal{Z}$. First, it is obvious that $\Phi(\cdot)$ is continuous if we choose *sigmoid* or *rectified* as non-linearity. It is also trivial to see that $\mathcal{Z}$ is a compact domain. Since $\mathcal{X}$ is compact as we only work on finite and bounded data, and that the encoder embedding is a deterministic function, the resulting $\mathcal{Z}$ is also finite, and therefore compact. We then see that $\Phi(\cdot)$ is now also bounded since continuous function on compact domain has to be bounded. Second, to show that RKHS is dense in $C(\mathcal{Z})$, recall that $\mathcal{F}$ is spanned by $\{\Phi_i = k(x_i, \cdot)\}_i$ which is a set of bounded and continuous functions. Then, the denseness of $\mathcal{F}$ in $C(\mathcal{Z})$ is immediately recognized by the well-known *universal approximation theorem* [11]. Finally, since k is universal, by Theorem 2, HSIC$(z, s) = 0$ if and only if z, s are independent. □

With the above claim, we can now safely apply HSIC with the kernel approximation method recognized by arbitrary deep neural network. We seek to minimize HSIC during training stage, and, as we have argued, the resulting representation will more likely to be independent of s by the above claim and Theorem 2. Unlike previous kernel approximation methods [25, 17] and the recently published deep kernel learning [28, 27], we do not fix a kernel and compute its approximation. Rather, we learn a new kernel $\Phi^T\Phi$ whenever we learn a new feature map. This is one major difference from previous kernel approaches and also one of our contributions. We have shown that feature maps parameterized by deep neural network and the kernel learned in this way can recognize any RKHS, and that we can use it as a criterion for independence measurement. The resulting objective would be easier to train as we can now project a batch of size m training examples by using $\Phi(\cdot)$, and the computational cost for HSIC only takes $O(m^2)$.

3.2 Model

We now introduce our model. As depicted by Figure 2, we learn a representation by encoding the input into feature space, and then we learn a neural network Φ to approximate a kernel. Φ is a function within some RKHS. Since we are given the factor s during training, the feature map Ψ for s is a fixed function, and is not learned. Since the protected factor s is discrete, we can simply use an indicator function. Specifically, we make Ψ a *one hot* embedding from z. We can get the associated kernel matrix by taking the outer product $K = \Phi\Phi^T$, $L = \Psi\Psi^T$. And then we can then use the empirical formula to compute HSIC.

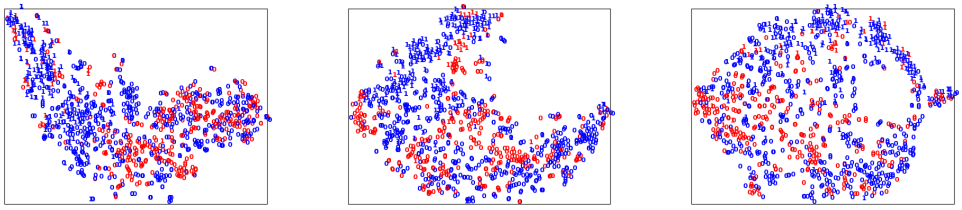
By our Claim 1, there is a one-to-one correspondence between z and s being statistically independent and HSIC(z, s) being zero. Therefore, we add HSIC as a constraint during training to learn a representation z and a feature map Φ in the prediction model such that HSIC(z, s) is small. To avoid the trivial solution, we add a *min-max* game in which the Φ's parameters are also learned to maximize HSIC. The resulting modified objective is therefore,

$$\min_{enc,f} \max_{\Phi} \mathcal{L}(x, y, s) = -\mathbb{E}_{z\sim enc(x)}[\log f(y|z)] + \lambda \cdot HSIC_{\Phi, z\sim enc(x)}(z, s)$$

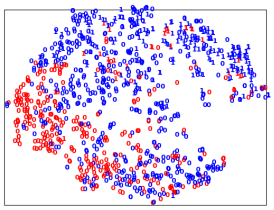
where λ controls the strength of the constraint. When Φ is zero everywhere, HSIC=0 even when z, s are not independent. The *min-max* game ensures that we will never learn a trivial Φ. Each time we learn a new Φ, by Claim 1 there will be an associated kernel and RKHS upon which we can calculate HSIC. The *max* game tries to make z and Φ to maximize HSIC, while the *min* game tries to learn invariant z. To see whether z has factored out unwanted information, after learning z, we will fix the

Table 1: Fair classification on Adult and German.

	Adult		German	
	Accuracy on s	**Accuracy on y**	**Accuracy on s**	**Accuracy on y**
Major line	0.67	0.75	0.78	0.70
2-layer NN (baseline)	0.78	0.88	0.78	0.91
VFAE [21]	0.67	0.84	0.78	0.72
Xie et al. [29]	0.67	0.83	0.78	0.70
Moyer et al. [23]	0.67	0.84	0.78	0.72
min-max COCO [9]	0.71	0.88	0.79	0.88
min-max HSIC (Ours)	**0.67**	**0.89**	**0.78**	**0.92**



(a) Baseline (b) Proposed, $\lambda = 1$ (c) Proposed, $\lambda = 10$



(d) COCO [9]

Figure 2: t-SNE of the learned latent features on Adult dataset (first 1000 samples of test data. Color indicates gender, for which blue presents blue represents male and red represents female. Target values are shown as 0 and 1.

encoder parameters, and then we try to learn a new classifier on top of the learned representation in order to predict sensitive factor s. If z does not contain much information of s, then it is expected that this classifier will always fail. We will discuss more details in the following section.

Table 2: Extended Yale B. Factoring out lighting information s. Results are from test data.

method	**Accuracy of classifying s**	**Accuracy of classifying y**
Original x	0.96	0.78
VFAE [21]	0.57	0.85
Xie et al. [29]	0.57	**0.89**
Ours	0.70	0.81

(a) baseline, $\lambda = 0$

(b) Proposed, $\lambda = 10$

(c) Proposed, $\lambda = 1000$

Figure 3: We conduct a similar style representation learning experiment as in [23]. The top row are the real images. The following rows are generated by varying class dependent code c. The goal is to make the general representation z which contains style information to be invariant to class c.

4 Experiments

4.1 Dataset and Experiment Setup

Fair Classification For fair classification, we used *Adult* and *German* datasets that are previously tested by other fair machine learning models [29, 21, 23]. Adult dataset contains 45222 number of samples of US census data. We use 50/50 split for training and testing, the same as previous studies. The objective is to predict whether a person makes more than $50k$ dollars a year. To achieve fair classification, we seek to learn a latent representation z that fit the classification problem while being invariant to sensitive attributes such as gender information. As there is internal bias in the dataset and many attributes other than gender are correlated with gender, simply dropping the gender attribute does not help to improve model fairness. Similarly, for *German* dataset, we seek to predict whether a person has good credit score while being fair with respect to protected factor like Age (which is binarized as in [31, 23]). It is a smaller dataset containing only 1000 data samples.

Invariant Representation Learning We use Extended Yale B [4, 20] from previous studies [29, 21] on learning invariant representations to examine our model performance on learning features that eliminates information of certain factors of variation. Extended Yale B contains images under 64 lighting conditions coming from 38 people. The objective is to predict the individual identity given an image while being invariant to illuminations. To factor out unwanted factor s, we select 5 groups of images with different light source [29, 21]: upper left, upper right, lower left, lower right, and front. There are $5 \times 38 = 190$ samples for training and all the other 2414 samples are left for testing.

Experiment Setup We use *Adam* [14] with learning rate 0.001 for training. We use *leaky ReLu* [5, 30] as non-linearity whenever possible. The hyperparameter HSIC coefficient λ is set to 10 for *Adult* and *German*, and 0.6 for Extended yale B. We train 30 epochs for fair classification, and train until the training loss converges for Extended Yale B. After we learn a latent feature z, we then train another classifier on top of z to predict sensitive factor s. The accuracy of this separate classifier is then used as a metric to determine whether the learned feature z has eliminate information of s. Ideally, any classifier on z should fail to predict s if z has factor any information about s. This strategy can also be found in previous studies [29].. We will provide full details of training, model architecture, and choice of hyperparameters in the Appendix (supplementary materials).

4.2 Results

4.2.1 Fair Classification

We report results for *Adult* and *German* datasets on fair classification. All results are based on Test data. Similar to [29], we include the accuracy of a separate classifier to see our model effectiveness on factoring our unwanted variations. The *Majority line* indicates the proportion of a variable in the dataset. For example, in *Adult* dataset, the unwanted factor s is gender, and there are 67% male in the population. In $German$ dataset, the unwanted factor is the binarized age, and there are 78% people younger than 44 in the population. The baseline is a 2 layer neural network with *ReLu* hidden units. It is clear that a simple neural network is successful on classifying the target, but there also exists a separate classifier which can predict sensitive attribute s with high accuracy. This indicates that the learned feature z contains information of s, and hence s can be easily inferred from z by another

classifier, which would lead to unfairness on the task of predicting target y, as the classifier for y will now take s into account indirectly. In *Adult* dataset, all fair machine learning models observe a drop of accuracy in terms of predicting s. It can be easily seen that there is a trade-off between fairness and model performance. A drop in accuracy on classifying s often comes with a drop in accuracy on classifying the actual target y. However, the comparison with baselines indicates that our model can successfully eliminate s while not sacrificing classification accuracy on y. We achieve the highest accuracy on y compared with previous ones. In comparison with the other kernel independence criterion, we also experiment with *covariance-correlation operator* COCO [9] by replacing the HSIC penality in our *min-max* game with COCO. As indicated by [9, 10], we found that HSIC works better empirically.

We also visualize the learned latent representation by t-SNE [22]. We project z onto a 2-dimensional space, see Figure 2. We label each sample point by the ground truth binarized target. Two different color represents male/female groups. For baseline, it is easy to see that points are clustered based on gender information, where all male samples tend to be on the left while female samples tent to be more on the right. It is clear that the latent features we learned do not have this issue. For large HSIC coefficient λ, samples from both genders form a similar sphere. z learned by using COCO is less affected by gender, but male/female are still separated compared with z learned by our proposed method. t-SNE visualization is also included in previous fair machine learning studies [21, 23].

4.2.2 Invariant Representation

For the more general problem of learning invariant representation, we use Extended yale B to test our model performance. The unwanted factor here is the 5 different lighting source s, and the target y is 38 person IDs. The baseline results show that the original representation x does not factor out s. An arbitrary classifier can be trained to have almost perfect accuracy on predicting the lighting source. However, as the test data containing a much larger amount of faces on 64 different lighting conditions, the classification accuracy on predicting IDs is not high. Learning representations invariant to s dramatically alleviates this issue. The same as previous studies, we observe a significant decrease in accuracy on s. And, since the representation is not invariant to s, the model achieves much higher accuracy even though there is a lot of new images generating from different s. We observe that the adversarial approach [29] is slightly better than our approach.The reason is not yet entirely clear to us. We suspect that one of is reason is related to the size of training data. There are only 190 samples for training. There could be other kernel approximation methods that perform better in such small dataset. Currently, most fair machine learning studies only work on small dataset. Using larger dataset is necessary in fair machine learning community. We will leave it for future research.

We also train a VAE with HSIC to learn style representation that is invariant to class information. Our conditional decoder is based on the concatenation of z and s, where s is the one hot class vector. Figure 3 shows our model performance on learning class invariant representation. We vary the class factor s while keeping z constant along the vertical line. It appears that z, s are more entangled for small λ and baselone VAE, especially for the first and last three columns.

5 Related Works

Our research is related to adversarial fair machine learning [29], in a way that we also include a *min-max* training objective. While they achieve high accuracy on face recognition, we achieve both better performance and fairness on two other baseline datasets. We also find that we are not the first one in fair machine learning community who make use of Hilbert Schmidt Independence Criterion. In fact, *Fair Kernel Learning* [24] is largely related to ours. However, there are two major differences. First, while they don't directly learn a representation, our goal is to learn an invariant representation that is not only useful for fair classification, but also for other general machine learning task such as style representation learning shown in Figure 3. Second, while their approach include a generalized eigenvalue problem, our approach makes use of *min-max* game and recent advancement of deep learning, which only results in $O(m^2)$ extra training time, where m is the size of a batch and is negligible. The simplicity of our method is one of the most appealing parts.

6 Conclusion

In this paper, we study the problem of fair and invariant representation learning. We propose a new and simple algorithm with Hilbert Schmidt Criterion that leads to fair model prediction. Unlike many other recent fair machine learning studies in which a trade-off between achieving fairness and model performance is often observed, our model tends to alleviate this trade-off. We combine HSIC, kernel learning, and deep neural network and give a concise and intuitive proof of our algorithm. We observe the performance drop in the human face classification dataset in which training data is 10 times smaller than test data. The problem of learning fair representation when there is not enough amount of data is intriguing. We will leave it for future studies.

References

[1] Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.

[2] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pp. 177–186. Springer, 2010.

[3] Andrey Feuerverger. A consistent test for bivariate dependence. *International Statistical Review/Revue Internationale de Statistique*, pp. 419–433, 1993.

[4] Athinodoros S Georghiades, Peter N Belhumeur, and David J Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6):643–660, 2001.

[5] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 315–323, 2011.

[6] 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*, pp. 2672–2680, 2014.

[7] Robert Kent Goodrich. A riesz representation theorem. *Proceedings of the American Mathematical Society*, 24(3):629–636, 1970.

[8] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pp. 63–77. Springer, 2005.

[9] Arthur Gretton, Ralf Herbrich, Alexander Smola, Olivier Bousquet, and Bernhard Schölkopf. Kernel methods for measuring independence. *Journal of Machine Learning Research*, 6(Dec): 2075–2129, 2005.

[10] Arthur Gretton, Kenji Fukumizu, Choon H Teo, Le Song, Bernhard Schölkopf, and Alex J Smola. A kernel statistical test of independence. In *Advances in neural information processing systems*, pp. 585–592, 2008.

[11] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.

[12] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. Fairness-aware learning through regularization approach. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pp. 643–650. IEEE, 2011.

[13] Konstantinos Kamnitsas, Christian Baumgartner, Christian Ledig, Virginia Newcombe, Joanna Simpson, Andrew Kane, David Menon, Aditya Nori, Antonio Criminisi, Daniel Rueckert, et al. Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. In *International conference on information processing in medical imaging*, pp. 597–609. Springer, 2017.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

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

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[17] Quoc Le, Tamás Sarlós, and Alex Smola. Fastfood-approximating kernel expansions in loglinear time. 2013.

[18] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[19] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[20] Kuang-Chih Lee, Jeffrey Ho, and David J Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (5):684–698, 2005.

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

[22] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[23] Daniel Moyer, Shuyang Gao, Rob Brekelmans, Aram Galstyan, and Greg Ver Steeg. Invariant representations without adversarial training. In *Advances in Neural Information Processing Systems*, pp. 9084–9093, 2018.

[24] Adrián Pérez-Suay, Valero Laparra, Gonzalo Mateo-García, Jordi Muñoz-Marí, Luis Gómez-Chova, and Gustau Camps-Valls. Fair kernel learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 339–355. Springer, 2017.

[25] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pp. 1177–1184, 2008.

[26] Saburou Saitoh. Theory of reproducing kernels and its applications. *Longman Scientific & Technical*, 1988.

[27] Andrew G Wilson, Zhiting Hu, Ruslan R Salakhutdinov, and Eric P Xing. Stochastic variational deep kernel learning. In *Advances in Neural Information Processing Systems*, pp. 2586–2594, 2016.

[28] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial Intelligence and Statistics*, pp. 370–378, 2016.

[29] Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. Controllable invariance through adversarial feature learning. In *Advances in Neural Information Processing Systems*, pp. 585–596, 2017.

[30] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.

[31] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pp. 325–333, 2013.