Fast and Efficient MMD-Based Fair PCA via Optimization over Stiefel Manifold

Junghyun Lee¹, Gwangsu Kim*², Mahbod Olfat³,⁴, Mark Hasegawa-Johnson⁵, Chang D. Yoo*²

¹ Kim Jae-chul Graduate School of AI, KAIST, Seoul, Republic of Korea
² School of Electrical Engineering, KAIST, Daejeon, Republic of Korea
³ UC Berkeley IEOR, Berkeley, CA, USA
⁴ Citadel, Chicago, IL, USA
⁵ Department of Electrical and Computer Engineering, University of Illinois Urbana-Champaign, IL, USA

{jh_lee00, s88012, cd_yoo}@kaist.ac.kr, molfat@berkeley.edu, jhasegaw@illinois.edu

Abstract

This paper defines fair principal component analysis (PCA) as minimizing the maximum mean discrepancy (MMD) between dimensionality-reduced conditional distributions of different protected classes. The incorporation of MMD naturally leads to an exact and tractable mathematical formulation of fairness with good statistical properties. We formulate the problem of fair PCA subject to MMD constraints as a non-convex optimization over the Stiefel manifold and solve it using the Riemannian Exact Penalty Method with Smoothing (REPMS). Importantly, we provide local optimality guarantees and explicitly show the theoretical effect of each hyperparameter in practical settings, extending previous results. Experimental comparisons based on synthetic and UCI datasets show that our approach outperforms prior work in explained variance, fairness, and runtime.

Introduction

It has become increasingly evident that many widely-deployed machine learning algorithms are biased, yielding outcomes that can be discriminatory across key groupings such as race, gender and ethnicity (Mehrabi et al. 2019). As the applications of these algorithms proliferate in protected areas like healthcare (Karan et al. 2012), hiring (Chien and Chen 2008) and criminal justice (Kirchner et al. 2016), this creates the potential for further exacerbating social biases. To address this, there has recently been a surge of interest in ensuring fairness in resulting machine learning algorithms.

Working in high-dimensional spaces can be undesirable as the curse of dimensionality manifests in the form of data sparsity and computational intractability. Various dimensionality reduction algorithms are deployed to resolve these issues, and Principal Component Analysis (PCA), is one of the most popular methods (Jolliffe and Cadima 2016). One particular advantage of PCA is that there’s no need to train a complex neural network.

In this work, fair PCA is defined as doing PCA while minimizing the difference in the conditional distributions of projections of different protected groups. Here, the projected data can be considered as a dimension-reduced fair representation of the input data (Zemel et al. 2013). We answer the questions of 1) how fairness should be defined for PCA and 2) how to (algorithmically) incorporate fairness into PCA in a fast and efficient manner. This work takes a different approach from prior studies on PCA fairness (Samadi et al. 2018; Olfat and Aswani 2019), which is discussed in Section 15.

Our main contributions are as follows:

- We motivate a new mathematical definition of fairness for PCA using the maximum-mean discrepancy (MMD), which can be evaluated in a computationally efficient manner from the samples while guaranteeing asymptotic consistency. Such properties were not available in the previous definition of fair PCA (Olfat and Aswani 2019). This is discussed in detail in Section 3 and 4.
- We formulate the task of performing MMD-based fair PCA as a constrained optimization over the Stiefel manifold and propose using REPMS (Liu and Boumal 2019). For the first time, we prove two general theoretical guarantees of REPMS regarding the local minimality and feasibility. This is discussed in detail in Section 5 and 6.
- Using synthetic and UCI datasets, we verify the efficacy of our approach in terms of explained variance, fairness, and runtime. Furthermore, we verify that using fair PCA does indeed result in a fair representation, as in (Zemel et al. 2013). This is discussed in detail in Section 7.

Preliminaries

Notations

For \( b \geq 1 \), let \( \mathcal{P}_b \) be the set of all Borel probability measures defined on \( \mathbb{R}^b \). For some measurable function \( \Pi: \mathbb{R}^p \rightarrow \mathbb{R}^d \) and a measure \( P \in \mathcal{P}_p \), the push-forward measure of \( P \) via \( \Pi \) is the probability measure \( \Pi_# P \in \mathcal{P}_d \), defined as \( (\Pi_# P)(S) = P(\Pi^{-1}(S)) \) for any Borel set \( S \). Let \( 0 \) and \( 1 \) denote matrices (or vectors) of zeros and ones of appropriate size, respectively. In this work, we focus on binary cases, i.e., we assume that the protected attribute \( A \) and outcome \( Y \) are binary (\( A, Y \in \{0,1\} \)). We abbreviate demographic parity, equalized opportunity, and equalized odds as DP, EOP, and EOD, respectively.
Maximum Mean Discrepancy (MMD)

Let $k: \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a positive-definite kernel function, and $H_k$ be a unit-ball in the RKHS generated by $k$. We impose some regularity assumptions on $k$:

**Assumption 1.** $k$ is measurable, and bounded i.e. $K := \sup_{x,y} k(x,y) < \infty$.

Then one can pseudo-metrize $\mathcal{P}_d$ by the following distance:

**Definition 1** (Gretton et al., 2007). Given $\mu, \nu \in \mathcal{P}_d$, their maximum mean discrepancy (MMD), denoted as $\text{MMD}_k(\mu, \nu)$, is a pseudo-metric on $\mathcal{P}_d$, defined as follows:

$$\text{MMD}_k(\mu, \nu) := \sup_{f \in H_k} \left| \int_{\mathbb{R}^d} f \, d(\mu - \nu) \right|$$  \hspace{1cm} (1)

As our fairness constraint involves exactly matching the considered distributions using MMD, we require the property of $\text{MMD}_k(\mu, \nu) = 0$ implying $\mu = \nu$. Any kernel $k$ that satisfies such property is said to be **characteristic** (Fukumizu et al. 2008) to $\mathcal{P}_d$. Furthermore, Sriperumbudur et al. (2008) defined and characterized stationary characteristic kernels and identified that well-known kernels such as RBF and Laplace are characteristic. Based on this fact, we set $k$ to be the RBF kernel $k_{\text{rbf}}(x, y) := \exp\left(-\|x - y\|^2/2\sigma^2\right)$.

For the choice of bandwidth $\sigma$, the median of the set of pairwise distances of the samples after vanilla PCA is considered following the **median heuristic** of (Schölkopf, Smola, and Müller 1998). For simplicity, we refer to $\text{MMD}_{k_{\text{rbf}}}$ as MMD.

**Benefits of MMD** There are several reasons for using MMD as the distance on a space of probability measures. First, it can act as a distance between distributions with different, or even disjoint, supports. This is especially crucial as the empirical distributions are often discrete and completely disjoint. Such a property is not generally true, one prominent example being the KL-divergence. Second, since many problems in fairness involve comparing two distributions, MMD has already been used in much of the fairness literature as a metric (Madras et al. 2018; Adel et al. 2019) and as an explicit constraint/penalty (Prost et al. 2019; Oneto et al. 2020; Jung et al. 2021), among other usages.

**Fairness for Supervised Learning**

The fair PCA discussed above should ultimately lead to fairness in supervised learning tasks based on the dimension-reduced data with minimal loss in performance. Let us now review three of the most widely-used definitions of fairness in supervised learning, as formulated in (Madras et al. 2018). Let $(Z, Y, A) \in \mathbb{R}^d \times \{0, 1\} \times \{0, 1\}$ be the joint distribution of the dimensionality-reduced data, (downstream task) label, and protected attribute. Furthermore, let $g : \mathbb{R}^d \to \{0, 1\}$ be a classifier that outputs prediction $\hat{Y}$ for $Y$ from $Z$. We want to determine the fairness of a well-performing classifier $g$ w.r.t. protected attribute $A$.

First, let $D_s$ be the probability measure of $Z_s \triangleq Z|A = s$ for $s \in \{0, 1\}$:

**Definition 2** (Feldman et al., 2015). $g$ is said to satisfy demographic parity (DP) up to $\Delta_{DP}$ w.r.t. $A$ with $\Delta_{DP} \triangleq |\mathbb{E}_{x \sim D_0}[g(x)] - \mathbb{E}_{x \sim D_1}[g(x)]|$.  

Now, let $D_{s,y}$ be the probability measure of $Z_s \triangleq Z|A = s, Y = y$ for $s, y \in \{0, 1\}$.

**Definition 3** (Hardt, Price, and Srebro, 2016). $g$ is said to satisfy equalized opportunity (EOP) up to $\Delta_{EOP}$ w.r.t. $A$ and $Y$ with $\Delta_{EOP} \triangleq |\mathbb{E}_{x \sim D_0}[g(x)] - \mathbb{E}_{x \sim D_1}[g(x)]|$.  

**Definition 4** (Hardt, Price, and Srebro, 2016). $g$ is said to satisfy equalized odds (EOD) up to $\Delta_{EOD}$ w.r.t. $A$ and $Y$ with $\Delta_{EOD} \triangleq \max_{y \in \{0, 1\}} |\mathbb{E}_{x \sim D_{0,y}}[g(x)] - \mathbb{E}_{x \sim D_{1,y}}[g(x)]|$.

From hereon, we refer to such $\Delta_f(g)$ as the **fairness metric of $f \in \{DP, EOP, EOD\}$ w.r.t. $g$, respectively.**

**New Definition of Fairness for PCA**

For $p > d$, let $\mathbb{R}^d$ be the space onto which data will be projected. A dimensionality reduction is a map $\Pi : \mathbb{R}^p \to \mathbb{R}^d$, and PCA is defined as $\Pi(x) = V^T x$ for some $V \in \mathbb{R}^{p \times d}$ satisfying $V V^T = I_d$ i.e. PCA is a linear, orthogonal dimensionality-reduction. From hereon, we denote a linear PCA as the mapping $\Pi$. The definition for fairness that we will be following throughout is given as follows.

**Definition 5** $(\Delta$-fairness). Let $P_s$ be the probability measure of $X_s \triangleq X|A = s$ for $s \in \{0, 1\}$, and let $Q_s := \Pi_{\#} P_s \in \mathcal{P}_d$. Then $\Pi$ is said to be $\Delta$-fair with $\Delta := \text{MMD}(Q_0, Q_1)$, and we refer to $\Delta$ as the **fairness metric**.

In other words, for lower $\Delta$-fairness, the discrepancy between the dimensionality-reduced conditional distributions of different protected classes, measured in a non-parametric manner using MMD, while retaining as much variance as possible, should be minimized.

Furthermore, Definition 5 ensures that a downstream classification task using $\Delta$-fair dimensionality-reduced data will be fair, as formalized below:

**Proposition 1** (Oneto et al., 2020). Up to a constant factor, $\text{MMD}(Q_0, Q_1)$ bounds the MMD of the push-forward measures of $Q_0, Q_1$ via the weight vector of any given downstream task classifier $g$.

**Remark 1.** The above discussions easily generalize to equal opportunity and equalized odds.

**Relation with Other Definitions of Fair PCA**

The notion of fairness proposed by Olfat and Aswani (2019) is similar to ours in that it measures the predictability of protected group membership in dimensionality-reduced data. However, unlike ours, their definition is explicitly adversarial, which can be a problem.

**Definition 6** $(\Delta_A$-fairness; Olfat and Aswani, 2019). Consider a fixed classifier $h(u, t) : \mathbb{R}^d \times \mathbb{R} \to \{0, 1\}$ that inputs

\begin{itemize}
  \item[1] The benefits of pursuing orthogonality in the loading matrix, and thus the resulting PCs, are already well-studied; for example, see (Qi, Luo, and Zhao 2013; Benidis et al. 2016).
  \item[2] See Lemma 3 of (Oneto et al. 2020) for the precise statement.
\end{itemize}
Moreover, for a family of classifiers \( \mathcal{F}_c \), if \( \Pi \) is \( \Delta_A(h) \)-fair for all \( h \in \mathcal{F}_c \), we say that \( \Pi \) is \( \Delta_A(\mathcal{F}_c) \)-fair.

As \( \Delta_A \) can’t be computed exactly, an estimator of the following form was used:

\[
\hat{\Delta}_A(\mathcal{F}_c) := \sup_{h \in \mathcal{F}_c} \sup_{t} \frac{1}{|P|} \sum_{i \in P} I_i(\Pi, h_t) - \frac{1}{|N|} \sum_{j \in N} I_j(\Pi, h_t)
\]

where \( \{x_i\}_{i=1}^n \) are the data points, \( (P, N) \) is a partition of the index set \( \{1, 2, \ldots, n\} \) into two protected groups, \( I_i(\Pi, h_t) = 1(h(\Pi(x_i), t) = +1) \), and \( 1(\cdot) \) is the indicator function.

**Remark 2.** It can be argued that, for some choice of \( \mathcal{F}_c \), Definition 5 and 6 are equivalent: in effect, that these are dual notions. Recognizing this, we proceed with Definition 5, as it has two main advantages in the context of our work:

- It ties more directly and intuitively into our optimization formulation; see Section .
- It can be represented non-variationally which allows for tighter statistical guarantees.

### Statistical Properties of \( \Delta \)

#### Consistent and Efficient Estimation of \( \Delta \)

As defined in Definition 5, let \( Q_0, Q_1 \in \mathcal{P}_d \) be the probability measures with respect to the samples of which we want to estimate \( \text{MMD}(\cdot, \cdot) \). Let \( \{X_i\}_{i=1}^m \) and \( \{Y_j\}_{j=1}^n \) be these samples, respectively. Accordingly, we consider the following estimator:

\[
\hat{\Delta} := \text{MMD}(\hat{Q}_0, \hat{Q}_1)
\]

where \( \hat{Q}_i \) is the usual empirical distribution, defined as the mixture of Dirac measures on the samples.

Unlike other statistical distances (e.g. total variation distance), \( \hat{\Delta}_k \) has several theoretical properties that have important practical implications; see Sriperumbudur et al. (2010) for more details.

First, it can be computed exactly and efficiently:

**Lemma 1** (Gretton et al., 2007). \( \hat{\Delta} \) is computed as follows:

\[
\hat{\Delta} = \left[ \frac{1}{m^2} \sum_{i,j=1}^m k(X_i, X_j) + \frac{1}{n^2} \sum_{i,j=1}^n k(Y_i, Y_j) - \frac{2}{mn} \sum_{i,j=1}^{m,n} k(X_i, Y_j) \right]^{1/2}.
\]

Moreover, it is asymptotically consistent with a convergence rate, depending only on \( m \) and \( n \):

**Proposition 2** (Gretton et al., 2007). For any \( \delta > 0 \), with probability at least \( 1 - 2 \exp \left( -\frac{\delta^2 mn}{2(m+n)} \right) \) the following holds:

\[
|\Delta - \hat{\Delta}| \leq 2 \left( \frac{1}{\sqrt{m}} + \frac{1}{\sqrt{n}} \right) + \delta
\]

### Advantages over \( \Delta_A \)

\( \hat{\Delta}_A \) is known to satisfy the following high probability bound:

**Proposition 3** (Olfat and Aswani, 2019). Consider a fixed family of classifiers \( \mathcal{F}_c \). Then for any \( \delta > 0 \), with probability at least \( 1 - \exp \left( -\frac{(m+n)\delta^2}{2} \right) \) the following holds:

\[
|\Delta_A(\mathcal{F}_c) - \hat{\Delta}_A(\mathcal{F}_c)| \leq 8 \sqrt{\frac{VC(\mathcal{F}_c)}{m+n}} + \delta
\]

where \( VC(\cdot) \) is the VC dimension.

**Remark 3.** If \( \mathcal{F}_c \) is too expressive in terms of VC-dimension, then the above bound may become void. This is the case, for instance, when \( \mathcal{F}_c \) is the set of RBF-kernel SVMs.

In addition, computing \( \hat{\Delta}_A \) requires considering all possible classifiers in the designated family \( \mathcal{F}_c \). This is computationally infeasible, and it forces one to use another approximation (e.g. discretization of \( \mathcal{F}_c \)), which incurs additional error that may further inhibit asymptotic consistency.

As exhibited in the previous subsection, our MMD-based approach suffers from none of these issues.

### Manifold Optimization for MBF-PCA

#### Improvements over FPCA

Olfat and Aswani (2019) proposed FPCA, an SDP formulation of fair PCA\(^3\), in which matching the first and second moments of the protected groups after dimensionality-reduction are approximated as convex constraints. However, this has several shortcomings, which we discuss here and empirically exhibit in a later section.

First, there are cases in which matching the mean and covariance alone is not enough. The simplest “counterexample” would be when two protected groups have the same mean and covariance, yet they have different distributions. This is illustrated in Figure 1. While this previous point can be countered by the application of the kernel trick to FPCA, this raises a second issue: their formulation requires solving \(^4\) a \( p \times p \)-dimensional SDP, motivated by the reparameterization \( P = V V^T \) (Arora, Cotter, and Srebro 2013). Since SDP is known to become inefficient (or even computationally infeasible) in high dimensions, this quickly becomes intractable for high-dimensional data (for linear or polynomial kernels) or for any moderate to large size datasets (for the RBF kernel).

Finally, their approach involves a relaxation of a rank constraint (\( \text{rank}(P) \leq d \)) to a trace constraint (\( \text{tr}(P) \leq d \)), yielding sub-optimal outputs in presence of (fairness) constraints, even to substantial order in some cases. In Section C

---

\(^3\)See Section B of the supplementary material for its precise description.

\(^4\)In their approach, the final solution \( V \) is obtained by taking the first \( d \) eigenvectors of \( P \).
of the SP, we discuss in detail why FPCA may lead to such degraded explained variance.

Formulating Mbf-PCA

Observing that the shortcomings of FPCA stem from the reparametrization of $P = V V^T$, we propose a new formulation of fair PCA that solves directly for $V$. This allows for an effective and efficient approach.

We start by noting that the set of all $V$ with orthonormal columns has the intrinsic geometric structure of a manifold:

**Definition 7.** For $p > d$, the Stiefel manifold, denoted as $St(p, d)$, is an embedded Riemannian sub-manifold of $\mathbb{R}^{p \times d}$ such that each element of $St(p, d)$ has orthonormal columns i.e. $V^T V = I_d$ for all $V \in St(p, d)$.

$St(p, d)$ has several desirable properties such as compactness, completeness and smoothness, which we present in Section D of the SP. As $St(p, d)$ is prevalent in various fields of machine learning (most notably PCA), much work has been done that focuses on exploiting this geometric structure for efficient optimization (Hu et al. 2020).

Based on our MMD-based formulation and letting $\Sigma$ be the sample covariance matrix of the full dataset, we formulate our fair PCA as follows, which we refer to as Mbf-PCA:

\[
\begin{align*}
\text{minimize} & \quad f(V) := -(\Sigma, V V^T) \\
\text{subject to} & \quad h(V) := \frac{1}{2} \text{MMD}(Q_0, Q_1) = 0.
\end{align*}
\]

Here, $Q_0$ and $Q_1$ are defined as in definition 5. Observe how our definition of fairness directly incorporates itself into the optimization problem as a constraint.

**Remark 4.** Under an assumption of normality, our MMD-based formulation amounts to the same constraints as FPCA since MMD is a metric and a Gaussian distribution is completely characterized by its first and second moments.

REPMS for Mbf-PCA, with New Theoretical Guarantees

**Description of Algorithm 1**

One crucial observation is that the constraint function $h$ is always non-negative and smooth. This motivates the use of the exact penalty method (Han and Mangasarian 1979), recently extended to manifold optimization as the Riemannian Exact Penalty Method via Smoothing (REPMS; Liu and Boumal, 2019). Note that smoothing tricks (Liu and Boumal 2019), which were required to smooth out possible non-differentiable functions emerging from the $\ell_1$-penalty, are not necessary. Moreover, by leveraging the kernel trick, there is a closed form for $\nabla V h(V)$, thus alleviating the need for computationally expensive numerical approximations; see Section G of the SP for the derivation. The pseudo-code for the algorithm is shown in Algorithm 1.

For practical concerns that will be addressed in the following subsection, we’ve set the fairness tolerance level, $\tau$, to be a fixed and sufficiently small, non-negative value. Formally, we consider the following definition:

**Definition 8.** For fixed $\tau \geq 0$, $V \in St(p, d)$ is $\tau$-approximate fair if it satisfies $h(V) \leq \tau$. If $\tau = 0$, we simply say that $V$ is fair.

**New Theoretical Guarantees for Algorithm 1**

We start by observing that Eq. (9) in Algorithm 1 is smooth, unconstrained manifold optimization problem, which can be solved using conventional algorithms; these include first-order methods like line-search methods (Absil, Mahony, and Sepulchre 2007), or second-order methods like the Riemannian Trust Region (RTR; Absil, Baker, and Gallivan, 2007) method. It is known that, pathological examples excluded, most conventional unconstrained manifold optimization solvers produce iterates whose limit points are local minima, and not other stationary points such as saddle point or local maxima: see (Absil, Baker, and Gallivan 2007; Absil, Mahony, and Sepulchre 2007) for more detailed discussions.

Motivated by this, we consider the following assumption:

---

5This is due to our choice of estimator for MMD inducing a non-negative estimate of MMD; see Section 2 of Gretton et al. (2012) for more detailed discussions.
Algorithm 1: REPMS for MbF-PCA

Input: $X$, $K$, $\epsilon_{\min}$, $\epsilon_0 > 0$, $\theta_2 \in (0, 1)$, $\rho_0 > 0$, $\theta_\rho > 1$, $\rho_{\max} \in (0, \infty)$, $\tau > 0$, $d_{\min} > 0$. 

1. Initialize $V_0$;
2. for $k = 0, 1, \ldots, K$ do
   3. Compute an approximate solution $V_{k+1}$ for the following sub-problem, with a warm-start at $V_k$, until $||\text{grad}\ Q|| \leq \epsilon_k$:
      \[
      \min_{V \in St(p, d)} Q(V, \rho_k)
      \]  
      where 
      \[
      Q(V, \rho_k) = f(V) + \rho_k h(V)
      \]
   4. if $\|V_{k+1} - V_k\|_F \leq d_{\min}$ and $\epsilon_k \leq \epsilon_{\min}$ then
      5. return $V_{k+1}$;
   6. end
   7. $\epsilon_{k+1} = \max\{\epsilon_{\min}, \theta_\epsilon \epsilon_k\}$;
   8. if $h(V_{k+1}) > \tau$ then
      9. $\rho_{k+1} = \min(\theta_\rho \rho_k, \rho_{\max})$;
   10. else
      11. $\rho_{k+1} = \rho_k$;
   12. end
13. end
14. end
15. end

Assumption 2 (informal; locality assumption). Each $V_{k+1}$ is sufficiently close to a local minimum of Eq. (9).

Lastly, we consider the following auxiliary optimization problem:

\[
\min_{V \in St(p, d)} h(V) \tag{10}
\]

The following theorem, whose proof is deferred to Section F of the SP, provides an exact theoretical convergence guarantee of MbF-PCA under the ideal hyperparameter setting:

**Theorem 1.** Let $K = \infty$, $\rho_{\max} = \infty$, $\epsilon_{\min} = \tau = 0$, $\{V_k\}$ be the sequence generated by Alg. 1 under Assumption 2, and $\bar{V}$ be any limit point of $\{V_k\}$, whose existence is guaranteed. Then the following holds:

(A) $\bar{V}$ is a local minimizer of Eq. (10), which is a necessary condition for $\bar{V}$ to be fair.

(B) If $\bar{V}$ is fair, then $\bar{V}$ is a local minimizer of Eq. (8).

The assumption of $\bar{V}$ being fair, which is used in (B), is at least partially justified in (A) in the following sense: the ideal hyperparameter setting of $\rho_{\max} = \infty, \tau = 0, \epsilon_{\min} = 0$ implies the exact local minimality of $\bar{V}$ for Eq. (10), which is in turn a necessary condition for $\bar{V}$ to be fair.

The next theorem, whose proof is also deferred to Section F of the SP, asserts that with small $\tau, \epsilon_{\min}$ and large $\rho_{\max}$, the above guarantee can be approximated in rigorous sense:

**Theorem 2.** Let $K = \infty$, $\rho_{\max} < \infty$, $\epsilon_{\min}, \tau > 0$, $\{V_k\}$ be the sequences generated by Alg. 1 under Assumption 2 and $\bar{V}$ be any limit point of $\{V_k\}$, whose existence is guaranteed. Then for any sufficiently small $\epsilon_{\min}$ and $\bar{r} = \bar{r}(\epsilon_{\min}) > 0$, the following hold:

(A) $\bar{V}$ is an approximate local minimizer of Eq. (10) in the sense that

\[
h(\bar{V}) \leq h(V) + \beta\|V - \bar{V}\| + (\beta + L_h)g(\epsilon_{\min}) \tag{11}
\]

for all $V \in B(\bar{V}) \cap St(p, d)$, where $\beta = \beta(\rho_{\max}, \tau)$ is a function that satisfies the following:

- $0 < \beta \leq \frac{2\|\Sigma\|}{\rho_0}$
- $\beta(\rho_{\max}, \tau)$ is increasing in $\rho_{\max}$ and decreasing in $\tau$.

(B) If $\bar{V}$ is fair, then it is an approximate local minimizer of Eq. (8) in the sense that it satisfies

\[
f(\bar{V}) \leq f(V) + 2\|\Sigma\|g(\epsilon_{\min}) \tag{12}
\]

for all fair $V \in B(\bar{V}) \cap St(p, d)$.

In both (A) and (B), $g$ is some continuous, decreasing function that satisfies $g(0) = 0$, and $\bar{r}(\epsilon_{\min}) = r - g(\epsilon_{\min})$ for some fixed constant $r > 0$.

Existing optimality guarantee of REPMS (Proposition 4.2; Liu and Boumal, 2019) states that when $\epsilon_{\min} = 0$, $\rho$ is not updated (i.e., line 10-14 is ignored), and the resulting limit point is feasible, then that limit point satisfies the KKT condition (Yang, Zhang, and Song 2014). Comparing Theorem 1 and 2 to the previous result, we see that ours extend the previous result in several ways:

- Our theoretical analyses are much closer to the actual implementation, by incorporating the $\rho$-update step (line 11) and the practical hyperparameter setting.
- Our theoretical analyses are much more stronger in the sense that 1) by introducing a reasonable, yet novel locality assumption, we go beyond the existing KKT conditions and prove the local minimality of the limit point, and 2) we provide a partial justification of the feasibility assumption in (A) by proving a necessary condition for it.

**Practical Implementation**

In line 4 in Algorithm 1, we implemented the termination criteria: sufficiently small distance between iterates and sufficiently small tolerance for solving Eq. (9). However, such a heuristic may return some point that is not $\tau$-approximate fair for user-defined level $\tau$ in practical hyperparameter setting. To overcome this issue, we’ve additionally implemented line 5 that forces the algorithm to continue on with the loop until the desired level of fairness is achieved.

**Related Work**

**Fairness in ML**

A large body of work regarding fairness in the context of supervised learning (Feldman et al. 2015; Calders, Kamiran, and Pechenizkiy 2009; Dwork et al. 2012; Hardt, Price, and Srebro 2016; Zafar et al. 2017) has been published. This includes key innovations in quantifying algorithmic bias, notably the concepts of demographic parity and equalized odds (opportunity) that have become ubiquitous in fairness research (Barocas and Selbst 2016; Hardt, Price, and Srebro 2016). More recently, fair machine learning literatures have branched out into a variety of fields, including deep learning...
(Beutel et al. 2017), regression (Calders et al. 2013), and even hypothesis testing (Olfat et al. 2020).

Among these, one line of research has focused on learning fair representations (Kamiran and Calders 2011; Zemel et al. 2013; Feldman et al. 2015; Calmon et al. 2017), which aims to learn a representation of the given data on which various fairness definitions are ensured for downstream modeling. A growing number of inquiries have been made into highly specialized algorithms for specific unsupervised learning problems like clustering (Chiierichetti et al. 2017; Bera et al. 2019), but these lack the general applicability of key dimensionality reduction algorithms such as PCA.

To the best of our knowledge, Olfat & Aswani (2019) is the only work on incorporating fair representation to PCA, making it the sole comparable approach to ours. Another line of work (Samadi et al. 2018; Tantipongpipat et al. 2019) considers a completely orthogonal definition of fairness for PCA: minimizing the discrepancy between reconstruction errors over protected attributes. This doesn’t ensure the fairness of downstream tasks, rendering it incomparable to our definition of fairness; see Section A of the SP for more details.

**Manifold Optimization**

A constrained problem over Euclidean space can be transformed to an unconstrained problem over a manifold (or at least manifold optimization with less constraints). Many algorithms for solving Euclidean optimization problems have direct counterparts in manifold optimization problems that includes Riemannian gradient descent and Riemannian BFGS. By making use of the geometry of lower dimensional manifold structure, often embedded in potentially very high dimensional ambient space, such Riemannian counterparts are much more computationally efficient than algorithms that do not make use of manifold structure. This is shown in numerous literatures (Liu and Boumal 2019; Alshairif et al. 2021), including this work. We refer interested readers to the standard textbooks Absil, Mahony, and Sepulchre (2007) and Boumal (2022) on this field, along with a survey by Hu et al. (2019).

**Experiments**

MbF-PCA was implemented using ROPTLIB (Huang et al. 2018), a state-of-the-art manifold optimization framework on MATLAB. For solving Eq. (9), we use the cautious Riemannian BFGS method (RBFGS; Huang, Absil, and Gallivan, 2018), a quasi-Newton method that is much more memory-efficient. We’ve set $K = 100$, $c_{\text{min}} = 10^{-6}$, $c_{\theta} = 10^{-1}$, $\theta_c = (c_{\text{min}}/c_{\theta})^{1/5}$, $\theta_{\text{max}} = 10^{10}$, $\theta_{\beta} = 2$, $d_{\text{min}} = 10^{-6}$. For FPCA, we use the same Python MOSEK/ApS (2021) implementation as provided by (Olfat and Aswani 2019). $(\mu, \delta)$ are the hyperparameters of FPCA; see Section B of the SP. Codes are available in our Github repository\(^6\).

All data is pre-processed to be standardized such that each covariate has zero mean and unit variance. For all experiments, we considered 10 different 70 – 30 train-test splits.

---

\(^6\)https://github.com/nick-jhlee/fair-manifold-pca

---

Figure 3: Synthetic data #2: Comparison of runtimes of FPCA, and MbF-PCA.

**Synthetic Data #1**

We consider synthetic data composed of two groups, each of size $n = 150$; one is sampled from $N_3(0, 0.1I_3 + 1)$ and one is sampled from a (balanced) mixture of $N_3(1, 0.1I_3)$ and $N_3(-1, 0.1I_3)$. Note how the two groups follow different distributions, yet have the same mean and covariance. Thus, we expect FPCA to project in a similar way as vanilla PCA, while MbF-PCA should find a fairer subspace such that the projected distributions are exactly the same. Hyperparameters are set as follows: $\delta = 0$, $\mu = 0.01$ for FPCA and $\tau = 10^{-5}$ for MbF-PCA. We’ve set $d = 2$ and Figure 1 displays the results of each algorithm using the top two principal components. Indeed, only MbF-PCA successfully obfuscates the protected group information by merging the two orange clusters with the blue cluster.

**Synthetic Data #2**

We consider a series of synthetic datasets of dimension $p$. For each $p$, the dataset is composed of two groups, each of size $n = 240$ and sampled from two different $p$-variate normal distributions. We vary $p \in \{20, 30, \ldots, 100\}$; see
Table 1: Comparison of PCA, FPCA, and MfB-PCA for UCI datasets. Number in parenthesis for each dataset is its dimension. Also, the parenthesis for each fair algorithm is its hyperparameter setting: \((\mu, \delta)\) for FPCA and \(\tau\) for MfB-PCA. Lastly, starred(*) results are those such that the maximum iteration is reached before passing the termination criteria in MfB-PCA.

Figure 4: Comparison of communality of “age” of German credit dataset for PCA, FPCA, and MfB-PCA.

Section H of the SP for a full description of the setting. For the hyperparameters, we’ve set \(\delta = 0, \mu = 0.01\) for FPCA and \(\tau = 10^{-3}\) for MfB-PCA.

Figure 2 plots the explained variance and fairness metric values. Observe how MfB-PCA achieves better explained variance, while achieving similar level of fairness. In addition, Figure 3 shows a clear gap in runtime between FPCA and MfB-PCA; the runtime of FPCA explodes for even moderate problem sizes, while MfB-PCA scales well. For higher dimensions, conventional computing machine will not be able to handle such computational burden.

UCI Datasets

For target dimensions \(d \in \{2, 10\}\), we compare the performance of FPCA and MfB-PCA on 3 datasets from the UCI Machine Learning Repository (Dufl and Graff 2017); COMPAS dataset (Kirchner et al. 2016); Adult income dataset, and German credit dataset. See Section I of the SP for complete description of the pre-processing steps. For both algorithms, we consider two different hyperparameter settings, such that one simulates the relaxed fairness while the other simulates a stricter fairness constraints. For computing \(\Delta_{DP}(g)\), we trained a RBF SVM \(g\) to be the downstream task classifier that best classifies the target attribute in the dimensionality-reduced data. Table 1 displays the results, in which among the fair algorithms considered, results with the best mean values are \textbf{bolded}. Several observations can be made:

- Across all considered datasets, MfB-PCA is shown to outperform FPCA in terms of fairness (both MMD\(^2\) and \(\Delta_{DP}\)) with low enough \(\tau\).
- For \textsc{German Credit} and \textsc{Adult Income}, MfB-PCA shows a clear trade-off between explained variance and fairness; by relaxing \(\tau\), we see that MfB-PCA outperforms FPCA in terms of explained variance and downstream task accuracy.

In addition, to see how correlated are the PCs with the protected attribute, we examine the communalities. For clarity of exposition, we consider the German credit dataset, whose protected attribute is age, and \(d = 10\). Here, we again consider PCA, FPCA (0, 0.01), and MfB-PCA (10\(^{-3}\)). For PCA, communality of a feature is its variance contributed by the PCs (Johnson and Wichern 2008), which is computed as the sum of squares of the loadings of the considered feature. Larger value of communality implies that the correlations between the considered feature and the PCs are strong. Figure 4 displays the boxplot of communality of considered 10 splits. Indeed the amount of variance in age that is accounted for from the loadings of MfB-PCA is much smaller than that of PCA or FPCA i.e. the PCs resulting from MfB-PCA have the least correlations with age, the protected attribute.

Conclusion and Future Works

We present a MMD-based definition of fair PCA, and formulate it as a constrained optimization over the Stiefel manifold. Through both theoretical and empirical discussions, we show that our approach outperforms the previous approach (Olfat and Aswani 2019) in terms of explained variance, fairness, and runtime. Many avenues remain for future research: statistical characterizations of our fair PCA in asymptotic regime, as well as incorporation of sparsity (Johnstone et al. 2009); incorporating stochastic optimization-type modifications (Shamir 2015), as such modifications are expected to result in better scalability and performance.

7369
Acknowledgements

This work was partly supported by Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2019-0-01396, Development of framework for analyzing, detecting, mitigating of bias in AI model and training data), and partly supported by Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (2021-0-01381, Development of Causal AI through Video Understanding).

References


