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

Junghyun Lee[1], Gwangsu Kim[2], Matt Olfat[3,4], Mark Hasegawa-Johnson[5], Chang D. Yoo[2]

[1]KAIST AI [2]KAIST EE [3]IEOR, UC Berkeley [4]Citadel [5]Dept. of ECE, UIUC

February 15, 2022

Outline

Fair Machine Learning

- An active area of research with enormous societal impact
  - cf. Machine Bias (Angwin et al., ProPublica 2016) - Black vs White Defendant's recidivism scores
- Machine learning algorithms should not be dependent on specific (sensitive) variables such as gender, age, race...etc.

	White	Black
Higher risk, yet didn't re-offend	23.5%	44.9%
Lower risk, yet did re-offend	47.7%	28.0%

Fair Machine Learning

- There are multiple frameworks on how to do this:
  - Fair supervised learning
  - Fair unsupervised learning
  - **Fair representation learning**
  - Fair data preprocessing
  - ...etc.
- Some useful resources:
  - `https://fairmlbook.org/pdf/fairmlbook.pdf`
  - `https://dl.acm.org/doi/pdf/10.1145/3457607`

Mathematically speaking, (in my humble opinion), **most** of the algorithmic fair ML problems can be formulated as *constrained optimizations*! (i.e. optimizationists(?)' roles are very important)

Fair Supervised Learning

- We briefly review three of the most widely-used definitions of fairness in supervised learning, as formulated in [MCPZ18].
- $(Z, Y, A) \in \mathbb{R}^d \times \{0, 1\} \times \{0, 1\}$: joint distribution of the dimensionality-reduced data, (downstream task) label, and protected attribute.
- $g : \mathbb{R}^d \to \{0, 1\}$: classifier that outputs prediction $\hat{Y}$ for Y from Z.
- D_s: probability measure of $Z_s \triangleq Z|A = s$ for $s \in \{0, 1\}$
- $D_{s,y}$: probability measure of $Z_s \triangleq Z|A = s, Y = y$ for $s, y \in \{0, 1\}$.

Fair Supervised Learning

Definition ([FFM+15])

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

Definition ([HPS16])

g is said to satisfy **equalized opportunity (EOP) up to** Δ_{EOP} w.r.t. A and Y with $\Delta_{EOP} \triangleq \left|\mathbb{E}_{x\sim D_{0,1}}[g(x)] - \mathbb{E}_{x\sim D_{1,1}}[g(x)]\right|$.

Definition ([HPS16])

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

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

Fair Supervised Learning

Actually used in legal literatures!

- Griggs v. Duke Power Co. (disparate impact)
  - "business hiring decision illegal if it resulted in disparate impact by race even if the decision was not explicitly determined based on race. The Duke Power Co. was forced to stop using intelligence test scores and high school diplomas, qualifications largely correlated with race, to make hiring decisions." [FFM+15]

While the Supreme Court has resisted a "rigid mathematical formula" defining disparate impact [20], we will adopt a generalization of the 80 percent rule advocated by the US Equal Employment Opportunity Commission (EEOC) [24]. We note that disparate impact itself is not illegal; in hiring decisions, business necessity arguments can be made to excuse disparate impact.

DEFINITION 1.1 (DISPARATE IMPACT ("80% RULE")). *Given data set $D = (X, Y, C)$, with* protected attribute *X (e.g., race, sex, religion, etc.), remaining attributes Y, and binary class to be predicted C (e.g., "will hire"), we will say that D has* disparate impact *if*

$$\frac{\Pr(C = YES \mid X = 0)}{\Pr(C = YES \mid X = 1)} \leq \tau = 0.8$$

for positive outcome class YES and majority protected attribute 1 where $\Pr(C = c \mid X = x)$ denotes the conditional probability (evaluated over D) that the class outcome is $c \in C$ given protected attribute $x \in X$.[1]

Fair Representation Learning

- Our work is in line with the fair representation learning [ZWS+13].
  - In our case, representation is a linear projection of the original data.
- "Representation learning is a promising approach for implementing algorithmic fairness" [CK19]
  - In this framework, a modular separation between roles can be made:
    - data regulator
    - data producer
    - data user
  - This has several positive implications:
    - centralize fairness constraints
    - simplify and centralize the task of fairness auditing
    - can be constructed to satisfy multiple fairness measures simultaneously
    - simplify the task of evaluating the fairness/performance tradeoff

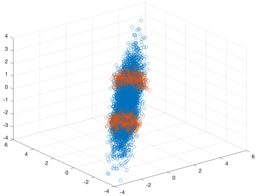
Problem Setting

- $\{x_i\}_{i=1}^n \subset \mathbb{R}^p$: original given data points (as row vectors)
  - $X \in \mathbb{R}^{n \times p}$: data matrix
  - Σ: empirical covariance matrix
- X is composed of two groups, which correspond to the protected classes (e.g. gender, age)
- $d < p$: dimension to which we want to reduce to
- $V \in \mathbb{R}^{p \times d}$: linear projection matrix (in case of PCA, $V^\intercal V = I_d$)
- Main objectives:

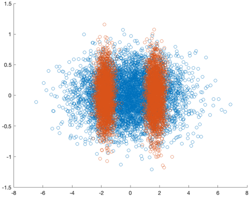
> - Maximize $\langle \Sigma, VV^\intercal \rangle$: *explained variance* of X after applying (linear) PCA using V.
> - Minimize fairness: *to be defined/discussed*

Problem setting

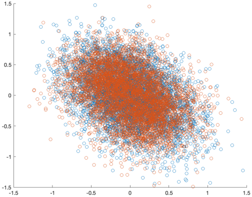
Fair PCA: the problem of maximizing the explained variance while imposing *distribution similarity after projection*!



(a) Original data



(b) Vanilla PCA



(c) Fair PCA

Outline

Adversarial Definition: FPCA

- To the best of our knowledge, [OA19] is the *only* prior work that considered this notion of fair PCA, in which they proposed the following adversarial definition:

Definition (Δ_A-fairness, [OA19] (Informal))

The dimensionality reduction $\Pi : \mathbb{R}^p \to \mathbb{R}^d$ is $\Delta_A(h)$-fair if adversarial classifiers that try to classify the protected class perform poorly in the projected space; the fairness metric is defined in terms of the difference between true positive and false positive.

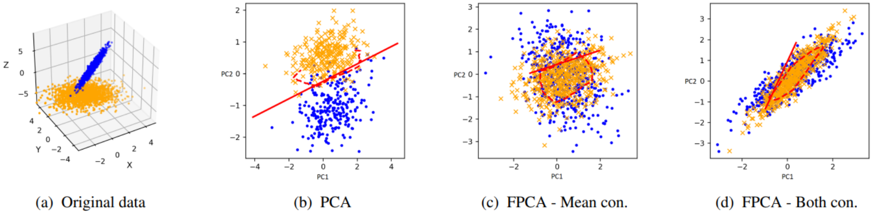


Figure 1: Comparison of PCA and FPCA on synthetic data. In each plot, the thick red line is the optimal linear SVM separating

SDP formulation of FPCA

- [OA19] provided a SDP formulation of above optimization[1]:

$$
\begin{aligned}
&\max \langle X^\intercal X, P\rangle - \mu t && \text{(7a)}\\
&\text{s.t. } \operatorname{trace}(P) \le d,\ \mathbb{I} \succeq P \succeq 0 && \text{(7b)}\\
&\langle P, ff^\intercal\rangle \le \delta^2 && \text{(7c)}\\
&\begin{bmatrix} t\mathbb{I} & PM_+ \\ M_+^\intercal P & \mathbb{I}\end{bmatrix} \succeq 0, && \text{(7d)}\\
&\begin{bmatrix} t\mathbb{I} & PM_- \\ M_-^\intercal P & \mathbb{I}\end{bmatrix} \succeq 0 && \text{(7e)}
\end{aligned}
$$

where $M_i M_i^\intercal$ is the Cholesky decomposition of $iQ + \varphi\mathbb{I}$ $(i \in \{-,+\})$, $\varphi \ge \|\widehat{\Sigma}_+ - \widehat{\Sigma}_-\|_2$, (7c) is called the *mean constraint* and denotes the use (5), and (7d) and (7e) are called the *covariance constraints* and are the SDP reformulation of (6). Our convex formulation for FPCA consists of solving (7) and then extracting the d largest eigenvectors from the optimal P^*.

Figure: δ: bound for mean difference, μ: bound for covariance difference

[1] This was heavily inspired from the SDP formulation of vanilla PCA [ACS13].

Problems with the Definition of FPCA

- Problems with the adversarial definition:
  - $\widehat{\Delta}_A(\mathcal{F})$, their fairness estimator, **cannot be computed exactly nor efficiently**.
  - Moreover, $\widehat{\Delta}_A(\mathcal{F})$ may be asymptotically **inconsistent**.
- Problems with the SDP algorithm:
  - The resulting solution is guaranteed to be **suboptimal** due to the SDP relaxations
  - As the fairness constraints were derived under **Gaussian assumption**, they do *not* ensure an exact distribution equality.
  - As the SDP is formulated w.r.t. $p \times p$ variable P, it is **inscalable** to high dimensions.

Computational Inefficiency

- Recall the definition of $\widehat{\Delta}_A$:

$$\widehat{\Delta}_A(\mathcal{F}_c) = \sup_{h \in \mathcal{F}_c} \sup_t \left| \frac{1}{|P|} \sum_{i \in P} I_i(\Pi, h_t) - \frac{1}{|N|} \sum_{i \in N} I_i(\Pi, h_t) \right|$$

- Computing above requires considering all possible classifiers in the designated family $\mathcal{F}_c$, and all possible thresholds $t \in \mathbb{R}$.
- 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.

Asymptotic Inconsistency

- $\widehat{\Delta}_A$ is known to satisfy the following bound:

Proposition ([OA19])

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

$$\left|\Delta_A(\mathcal{F}_c) - \widehat{\Delta}_A(\mathcal{F}_c)\right| \leq 8\sqrt{\frac{VC(\mathcal{F}_c)}{m+n}} + \delta \tag{1}$$

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

- If $\mathcal{F}_c$ is too expressive, then the above bound may become void!
- This is the case, for instance, when $\mathcal{F}_c$ is the set of RBF-kernel SVMs, whose VC dimension is infinite...

Always suboptimal

- The orthogonality constraint $V^\intercal V = I_d$ was *relaxed* to the trace bound and some matrix inequalities (especially, $\mathrm{rank}(P) \leq d \Rightarrow \mathrm{tr}(P) \leq d$, where P is an auxiliary SDP matrix variable)
- *Without* the fairness constraints, such relaxation can be proven to be exact [OA19].
- *With* the fairness constraints, such relaxation guarantees suboptimality...

Inexact fairness

- The fairness constraints for the SDP were derived under the assumption that the underlying datas are *Gaussian*.
- This does *not* cover the general distributional equality... (i.e. there exists two distributions that are different, yet have the same first and second moments.)

Inscalable to high dimensions

- Recall that the SDP is solved w.r.t. a new variable, $P \in \mathbb{R}^{p \times p}$, where $P = VV^\intercal$.
- Recall that p is the original data's dimension...
- i.e. we expect the time complexity to scale polynomially w.r.t. the original data's dimension p, *irrespective* of the dimension d to which we are reducing to!

Outline

Maximum mean discrepancy (MMD)

- It's clear that we need a new definition of fairness in PCA that can
  - directly lead to a tractable and exact optimization
  - be interpreted more easily and more intuitively
- We use the notion of MMD:

Definition ([GBR+07])

Given $\mu, \nu \in \mathcal{P}_d$, their **maximum mean discrepancy (MMD)**, denoted as $MMD_k(\mu, \nu)$, is a pseudo-metric on $\mathcal{P}_d$, defined as follows:

$$MMD_k(\mu, \nu) := \sup_{f \in \mathcal{H}_k} \left| \int_{\mathbb{R}^d} f \; d(\mu - \nu) \right|$$

- $\mathcal{P}_d$ is the set of all possible probability measures defined on $\mathbb{R}^d$.
- With characteristic kernels [FGSS08] such as the RBF kernel, MMD_k becomes a *metric* on $\mathcal{P}_d$.
  - From hereon and forth, we only consider MMD with the RBF kernel.

Δ-fairness

- Motivated from previous discussions, we propose a new definition for fair PCA based on MMD:

Definition (Δ-fairness (informal))

The dimensionality reduction $\Pi : \mathbb{R}^p \to \mathbb{R}^d$ is Δ-fair with Δ being the MMD of projected distributions, which is precisely the fairness metric.

- Well-known properties of MMD [GBR+07] already make it superior over the previous adversarial definition:

  - $\widehat{\Delta}$ can be computed exactly and efficiently.
  - $\widehat{\Delta}$ is asymptotically consistent.
  - As it is a metric over $\mathcal{P}_d$, no assumption on the datas has to be made; $MMD = 0$ is itself the fairness constraint!

Computational Efficiency

- We consider the following estimator:

$$\widehat{\Delta} := MMD(\hat{Q}_0, \hat{Q}_1) \tag{2}$$

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

- Unlike $\widehat{\Delta}_A$, $\widehat{\Delta}$ can be computed exactly and efficiently:

Lemma ([GBR+07])

$\widehat{\Delta}$ is computed as follows:

$$\widehat{\Delta} = \left[\frac{1}{m^2} \sum_{i,j=1}^{m} k(X_i, X_j) + \frac{1}{n^2} \sum_{i,j=1}^{n} k(Y_i, Y_j) - \frac{2}{mn} \sum_{i,j=1}^{m,n} k(X_i, Y_j) \right]^{1/2}. \tag{3}$$

Asymptotic Consistency

- Unlike $\widehat{\Delta}_A$, $\widehat{\Delta}$ is asymptotic convergent, with the rate depending only on m and n:

Theorem ([GBR[+]07])

For any $\delta > 0$, with probability at least $1 - 2\exp\left(-\frac{\delta^2 mn}{2(m+n)}\right)$ the following holds:

$$\left|\Delta - \widehat{\Delta}\right| \leq 2\left(\frac{1}{\sqrt{m}} + \frac{1}{\sqrt{n}}\right) + \delta \tag{4}$$

Optimization Formulation of Fair PCA

- All of the aformentioned problems of FPCA originated from the approach that *the optimization was not directly w.r.t. V*
  - The SDP formulation of FPCA was w.r.t. $P = VV^\intercal \in \mathbb{R}^{p \times p}$; the final solution is obtained by the eigendecomposition of the resulting P^*.

> Instead of trying to transform our problem into some surrogate optimization problem (ex. SDP), let us optimize **directly** for V!

$$
\begin{aligned}
\underset{V \in \mathbb{R}^{p \times d}}{\text{maximize}} \quad & \langle \Sigma, VV^\intercal \rangle \\
\text{subject to} \quad & V^\intercal V = \mathbb{I}_d, \\
& h(V) := MMD^2(\hat{Q}_0, \hat{Q}_1) = 0.
\end{aligned}
\tag{5}
$$

- Above is a smooth, nonconvex optimization.

Fair PCA as Manifold Optimization

- We utilize the *manifold structure of PCA*, namely, that the set of all V's with $V^\intercal V = \mathbb{I}_d$ forms the Stiefel manifold, denoted as $St(p, d)$.
- Then the previous problem can be formulated as a constrained manifold optimization problem, which we refer to as MbF-PCA:

$$\begin{aligned} \underset{V \in St(p,d)}{\text{maximize}} \quad & \langle \Sigma, VV^\intercal \rangle \\ \text{subject to} \quad & h(V) := MMD^2(\hat{Q}_0, \hat{Q}_1) = 0. \end{aligned} \tag{6}$$

- This has several advantages:

  - No relaxation was required, which means that theoretically, global minimizer of above is precisely the optimal solution.
  - Manifold optimization helps with the inscalability in high dimensions as the low-dimensional embedded geometry is used.

Quick Intuition behind Manifold Optimization

- Consider $\mathcal{M}$, an embedded Riemannian sub-manifold of $\mathbb{R}^{p \times d}$.
- Suppose we want to minimize some function $f : \mathbb{R}^{p \times d} \to \mathbb{R}$ *over* $\mathcal{M}$.
- If $\mathcal{M}$ is simply viewed as a subset of $\mathbb{R}^{p \times d}$, then this is a constrained optimization problem:

$$\begin{aligned} \underset{V}{\text{minimize}} \quad & f(V) \\ \text{subject to} \quad & V \in \mathcal{M}. \end{aligned} \tag{7}$$

- In this case, the optimization algorithm will make use of the canonical gradients and Hessians of $\mathbb{R}^{p \times d}$.

Quick Intuition behind Manifold Optimization

- If $\mathcal{M}$ is "all there is", then this problem is an unconstrained optimization problem over $\mathcal{M}$.
  - Consider an ant living on $\mathcal{M}$. From the universe ($\mathbb{R}^{p \times d}$), the ant is constrained on $\mathcal{M}$. But from the ant's perspective, $\mathcal{M}$ is all they have i.e. he/she would feel *unconstrained*!
- In this case, the optimization algorithm will make use of the *Riemannian* gradients and Hessians of $\mathcal{M}$.
- By making use of the intrinsic geometry of $\mathcal{M}$, the optimization becomes much more efficient!

Quick Intuition behind Manifold Optimization

- A very straightforward way to think of this is by considering the simplest Riemannian manifold[2], $\mathbb{R}^{p \times d}$.
- When we write the optimization as

$$\begin{aligned} \underset{V}{\text{minimize}} \quad & f(V) \\ \text{subject to} \quad & V \in \mathbb{R}^{p \times d}, \end{aligned} \tag{8}$$

  technically this is a "constrained" optimization because we're "constraining" V to be in $\mathbb{R}^{p \times d}$.
- However, gradients and Hessian (and other geometric concepts) are derived directly from the intrinsic geometry of $\mathbb{R}^{p \times d}$ i.e. $V \in \mathbb{R}^{p \times d}$ **isn't considered as a constraint.**

[2] inner product is the Frobenius product: $\langle X, Y \rangle := \mathrm{tr}(X^\intercal Y)$

REPMS for MbF-PCA

- To solve the optimization, we use REPMS [LB19], a Riemannian counterpart for the exact penalty method:

Algorithm 1: REPMS for MbF-PCA

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

Initialize V_0;
for $k = 0, 1, \ldots, K$ **do**
  Compute an approximate solution V_{k+1} for the following sub-problem, with a warm-start at V_k, until $\|\text{grad } \mathcal{Q}\| \leq \epsilon_k$:

$$\min_{V \in St(p,d)} \mathcal{Q}(V, \rho_k) \tag{9}$$

  where

$$\mathcal{Q}(V, \rho_k) = f(V) + \rho_k h(V)$$

  if $\|V_{k+1} - V_k\|_F \leq d_{min}$ *and* $\epsilon_k \leq \epsilon_{min}$ **then**
    if $h(V_{k+1}) \leq \tau$ **then**
      return V_{k+1};
    end
  end
  $\epsilon_{k+1} = \max\{\epsilon_{min}, \theta_\epsilon \epsilon_k\}$;
  if $h(V_{k+1}) > \tau$ **then**
    $\rho_{k+1} = \min(\theta_\rho \rho_k, \rho_{max})$;
  else
    $\rho_{k+1} = \rho_k$;
  end
end

Figure: Pseudocode of REPMS

New Theoretical Guarantees

- Our problem is non-convex in V, which naturally brings up the question of convergence and optimality guarantees.
- First, from various Riemannian optim literatures, we motivate the following assumption, which is to the best of our knowledge, new:

Assumption (informal; locality assumption)

Each V_{k+1} is sufficiently close to a local minimum of Eq. (9).

- 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 [ABG07, AMS07] for more detailed discussions.
- Many theoretical results have also emerged (ex. "First-order methods almost always avoid strict saddle points" Lee et al., Math. Prog. 2019)

New Theoretical Guarantees

- Under some mild conditions (see the paper for more details), we derive two *new* theoretical guarantees for REPMS.

Theorem

Let $K = \infty$, $\rho_{max} = \infty$, $\epsilon_{min} = \tau = 0$, $\{V_k\}$ be the sequence generated by REPMS, and $\overline{V}$ be any limit point of $\{V_k\}$, whose existence is guaranteed. Then the following holds:

- *$\overline{V}$ always satisfies a necessary condition for $\overline{V}$ to be fair.*
- *If $\overline{V}$ is fair, then $\overline{V}$ is a local maximizer of Eq. (6)*

Theorem (Informal)

Let $K = \infty$, $\rho_{max} < \infty$, $\epsilon_{min}, \tau > 0$. Then above holds approximately in the following sense: as $\rho_{max} \to \infty$ and $\epsilon_{min}, \tau \to 0$, we recover the previous exact guarantees.

Novelty of our theoretical guarantees

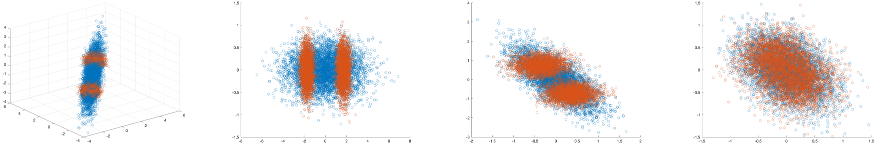
- Existing optimality guarantee of REPMS (Proposition 4.2; [LB19]):
  - $\epsilon_{min} = \tau = 0$, ρ is *not* updated (i.e. line 10-14 is ignored)
  - If the resulting limit point is fair, then that limit point satisfies the Riemannian KKT condition [YZS14].
- Our theoretical analyses[3]:
  - $\epsilon_{min}, \tau \geq 0$, ρ *is* updated
  - If the resulting limit point is (approximately) fair, then that limit point is (approximately) local maximizer.

[3] We've incorporated a new, yet reasonable assumption; see our paper for more details.

Outline

Synthetic data #1

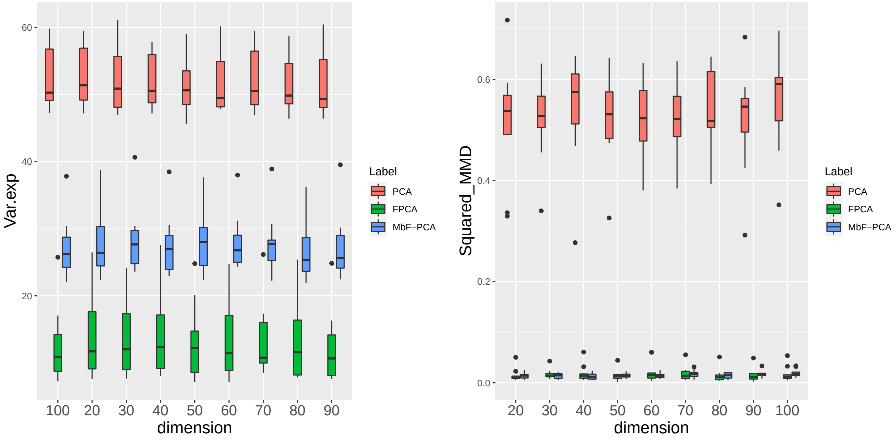
- Due to the Gaussian assumption, FPCA cannot cover the case when two sensitive distributions, that are different, have the same first two moments (mean, covariance):



(a) Original data (b) PCA (c) FPCA [OA19] (d) MbF-PCA (ours)

Figure: Synthetic data #1: Comparison of PCA, FPCA, and MbF-PCA on data composed of two groups with same mean and covariance, but different distributions. Blue and orange represent different protected groups.

Synthetic data #2



(a) Variance explained (%) (b) MMD^2

Figure: Synthetic data #2: Comparison of PCA, FPCA, and MBF-PCA on the synthetic datasets of increasing dimensions.

Synthetic data #2

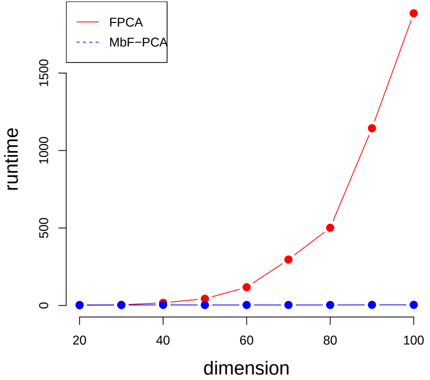


Figure: FPCA represents the SDP algorithm for fair PCA, and MbF-PCA represents our manifold-based framework. Note the drastic difference in scalability!

UCI Datasets

Table 1: Comparison of PCA, FPCA, MBF-PCA for UCI datasets. Number in parenthesis for each dataset is its dimension. Also, the parenthesis for each fair algorithm is its hyperparameter setting; (μ, δ) for FPCA and τ for MBF-PCA. Among the fair algorithms considered, results with the best mean values are **bolded**. Results in which our approach terminates improperly in the sense that the maximum iteration is reached before passing the termination criteria are highlighted.

d	Alg.	COMPAS (11)				German Credit (57)				Adult Income (97)			
		%Var	%Acc	MMD^2	Δ_{DP}	%Var	%Acc	MMD^2	Δ_{DP}	%Var	%Acc	MMD^2	Δ_{DP}
2	PCA	$39.28_{5.17}$	$64.53_{1.45}$	$0.092_{0.010}$	$0.29_{0.09}$	$11.42_{0.47}$	$76.87_{1.39}$	$0.147_{0.049}$	$0.12_{0.06}$	$7.78_{0.82}$	$82.03_{1.15}$	$0.349_{0.027}$	$0.20_{0.05}$
	FPCA $(0.1, 0.01)$	$\mathbf{35.06_{5.16}}$	$61.65_{1.17}$	$0.012_{0.007}$	$0.10_{0.07}$	$7.43_{0.59}$	$72.17_{1.09}$	$0.017_{0.010}$	$0.03_{0.02}$	$4.05_{0.98}$	$77.44_{2.96}$	$0.016_{0.011}$	$0.04_{0.04}$
	FPCA $(0, 0.01)$	$34.43_{5.02}$	$60.86_{1.09}$	$0.011_{0.006}$	$0.10_{0.06}$	$7.33_{0.57}$	$71.77_{1.60}$	$\mathbf{0.015_{0.010}}$	$0.03_{0.03}$	$3.65_{0.97}$	$77.05_{3.18}$	$\mathbf{0.005_{0.004}}$	$\mathbf{0.01_{0.01}}$
	MBF-PCA (10^{-3})	$33.95_{5.01}$	$\mathbf{65.37_{1.11}}$	$0.005_{0.002}$	$0.12_{0.07}$	$\mathbf{10.17_{0.57}}$	$\mathbf{74.53_{1.92}}$	$0.018_{0.014}$	$0.05_{0.04}$	$\mathbf{6.03_{0.61}}$	$\mathbf{79.50_{1.22}}$	$\mathbf{0.005_{0.004}}$	$0.03_{0.02}$
	MBF-PCA (10^{-6})	$11.83_{3.59}$	$57.73_{1.50}$	$\mathbf{0.002_{0.002}}$	$\mathbf{0.06_{0.08}}$	$9.36_{0.33}$	$74.10_{1.56}$	$0.016_{0.010}$	$\mathbf{0.02_{0.02}}$	$5.83_{0.57}$	$79.12_{1.14}$	$\mathbf{0.005_{0.004}}$	$\mathbf{0.01_{0.01}}$
10	PCA	$100.00_{0.00}$	$73.14_{1.22}$	$0.241_{0.005}$	$0.21_{0.07}$	$38.25_{0.98}$	$99.93_{0.14}$	$0.130_{0.019}$	$0.12_{0.08}$	$21.77_{2.06}$	$93.64_{0.92}$	$0.195_{0.007}$	$0.16_{0.01}$
	FPCA $(0.1, 0.01)$	$\mathbf{87.79_{1.27}}$	$72.25_{0.93}$	$0.015_{0.003}$	$\mathbf{0.16_{0.06}}$	$29.85_{0.87}$	$\mathbf{99.93_{0.14}}$	$0.020_{0.005}$	$0.12_{0.08}$	$15.75_{1.20}$	$91.94_{0.88}$	$0.006_{0.003}$	$0.13_{0.02}$
	FPCA $(0, 0.1)$	$87.44_{1.35}$	$\mathbf{72.32_{0.93}}$	$0.015_{0.002}$	$\mathbf{0.16_{0.07}}$	$29.79_{0.89}$	$\mathbf{99.93_{0.14}}$	$0.020_{0.006}$	$0.12_{0.08}$	$15.52_{1.18}$	$91.66_{0.97}$	$0.004_{0.002}$	$0.13_{0.02}$
	MBF-PCA (10^{-3})	$87.75_{1.36}$	$72.16_{0.90}$	$\mathbf{0.014_{0.002}}$	$\mathbf{0.16_{0.07}}$	$\mathbf{34.10_{1.00}}$	$\mathbf{99.93_{0.14}}$	$0.020_{0.008}$	$0.12_{0.08}$	$\mathbf{18.71_{1.47}}$	$\mathbf{92.81_{0.84}}$	$0.005_{0.002}$	$0.14_{0.01}$
	MBF-PCA (10^{-6})	$87.75_{1.36}$	$72.16_{0.90}$	$\mathbf{0.014_{0.002}}$	$\mathbf{0.16_{0.07}}$	$16.95_{1.52}$	$92.70_{3.00}$	$\mathbf{0.013_{0.007}}$	$\mathbf{0.06_{0.05}}$	$15.49_{6.44}$	$86.36_{3.77}$	$\mathbf{0.003_{0.002}}$	$\mathbf{0.07_{0.03}}$

- Across all considered datasets, MBF-PCA is shown to outperform FPCA in terms of fairness (MMD^2 and Δ_{DP}) with low enough τ.
- For German Credit and Adult Income, MBF-PCA shows a clear trade-off between explained variance and fairness:
  - By relaxing τ, MBF-PCA outperforms FPCA in terms of explained variance and downstream task accuracy, and vice-versa.

UCI Datasets

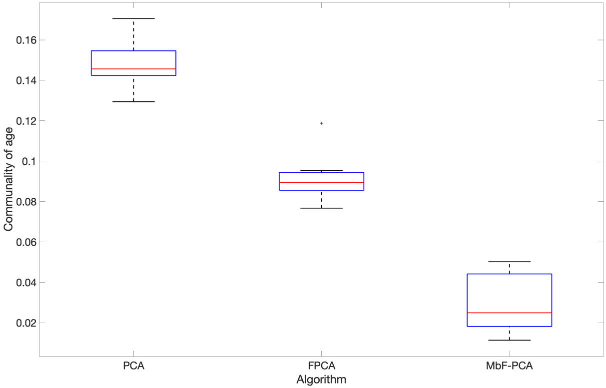


Figure: Comparison of communality of "age" of German credit dataset for PCA, FPCA, and MbF-PCA.

Conclusion

Our contributions:

- MbF-PCA: a new framework for fair PCA
  - **New definition** for fair PCA based on MMD, which is better than the previous definition [OA19].
  - Utilization of **manifold optimization framework** for MbF-PCA, which is also better than the previous SDP-based approach [OA19].
- **New optimality guarantees** for REPMS, extending [LB19].
- Empirical verification of our algorithm on synthetic and UCI datasets in explained variance, fairness, and runtime.

Check out our paper for more details! (and come to our poster for more discussions!)

Paper: `https://arxiv.org/abs/2109.11196`
Github: `https://github.com/nick-jhlee/fair-manifold-pca`

P.-A. Absil, C. G. Baker, and K. A. Gallivan, *Trust-region methods on riemannian manifolds*, Foundations of Computational Mathematics **7** (2007), no. 3, 303–330.

Raman Arora, Andy Cotter, and Nati Srebro, *Stochastic optimization of pca with capped msg*, Advances in Neural Information Processing Systems (C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), vol. 26, Curran Associates, Inc., 2013, pp. 1815–1823.

P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*, Princeton University Press, USA, 2007.

Moustapha Cisse and Sanmi Koyejo, *Nips 2019 tutorial: Fairness and representation learning*, 2019.

Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian, *Certifying and removing disparate impact*, Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Sydney, NSW, Australia), 2015, pp. 259–268.

Kenji Fukumizu, Arthur Gretton, Xiaohai Sun, and Bernhard Schölkopf, *Kernel measures of conditional dependence*, Advances in Neural Information Processing Systems (J. Platt, D. Koller, Y. Singer, and S. Roweis, eds.), vol. 20, Curran Associates, Inc., 2008.

Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola, *A kernel method for the two-sample-problem*, Advances in Neural Information Processing Systems (B. Schölkopf, J. Platt, and T. Hoffman, eds.), vol. 19, MIT Press, 2007.

Moritz Hardt, Eric Price, and Nati Srebro, *Equality of opportunity in supervised learning*, Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016 (Barcelona, Spain), 2016, pp. 3315–3323.

Changshuo Liu and Nicolas Boumal, *Simple algorithms for optimization on riemannian manifolds with constraints*, Applied Mathematics and Optimization **82** (2019), 949–981.

David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel, *Learning adversarially fair and transferable representations*, Proceedings of the 35th International Conference on Machine Learning (Jennifer Dy and Andreas Krause, eds.), Proceedings of Machine Learning Research, vol. 80, PMLR, 10–15 Jul 2018, pp. 3384–3393.

Matt Olfat and Anil Aswani, *Convex formulations for fair principal component analysis*, The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, 2019, pp. 663–670.

Wei Hong Yang, Lei-Hong Zhang, and Ruyi Song, *Optimality conditions for the nonlinear programming problems on riemannian manifolds*, Pacific Journal of Optimization **10** (2014), 415–434.

Richard S. Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork, *Learning fair representations*, Proceedings of the 30th International Conference on Machine Learning, ICML 2013 (Atlanta, GA, USA), 2013, pp. 325–333.