Fair dimensionality reduction and iterative rounding for SDPs

UTHAIPON (TAO) TANTIPONGPIPAT

GEORGIA INSTITUTE OF TECHNOLOGY

JOINT WORK WITH

JAMIE MORGERNSTERN, SAMIRA SAMADI, MOHIT SINGH, AND SANTOSH VEMPALA

Outline

The problem

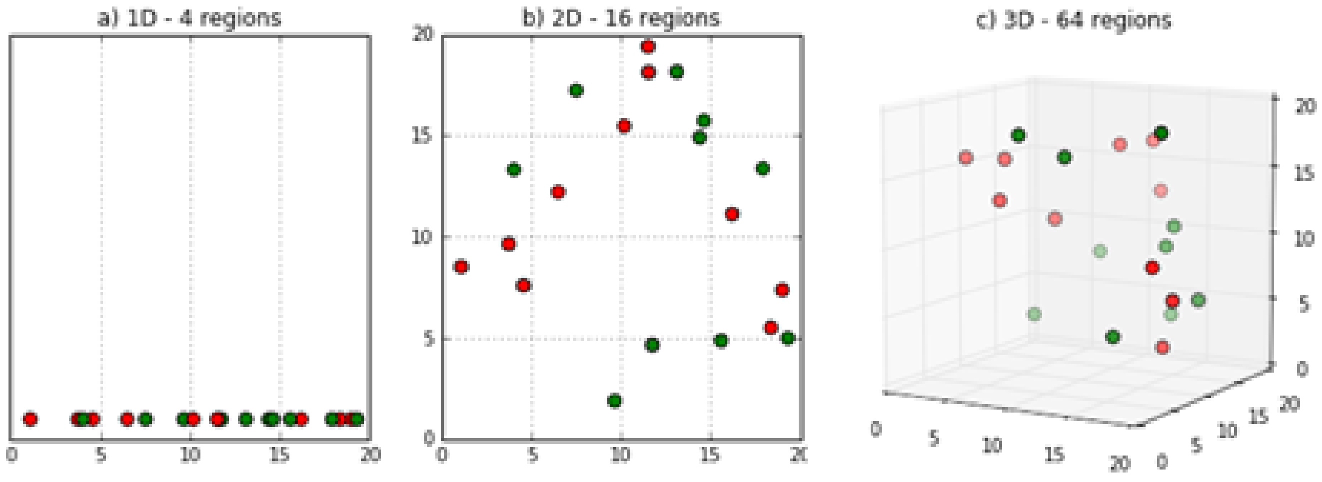
- Motivation → Define fair PCA and fair dimensional reduction (fair DM)

The method

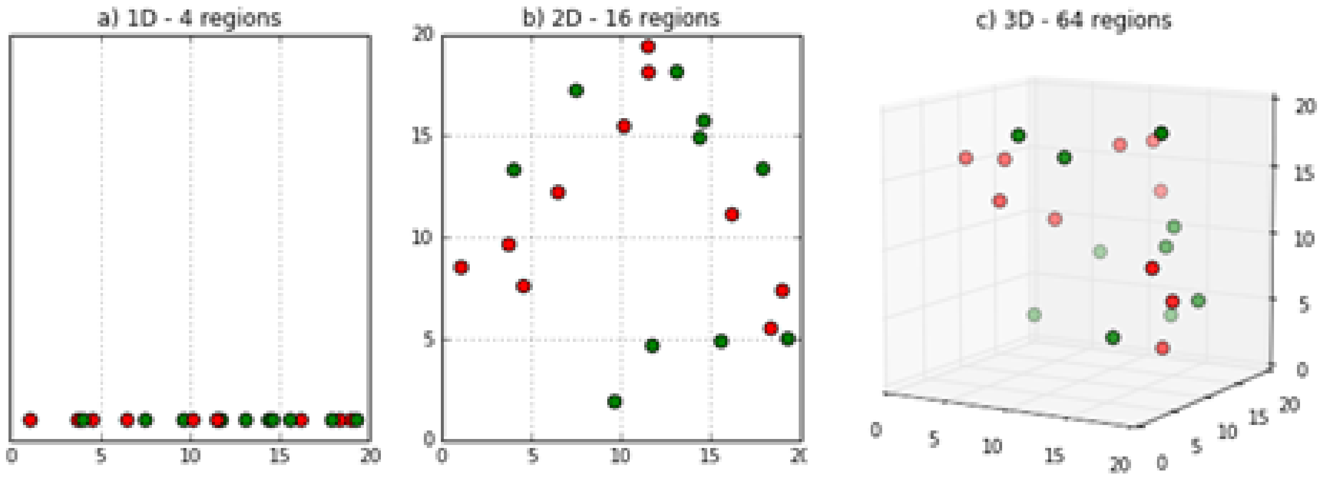
- Two of our algorithms (focus on one) and their analysis

Conclusion

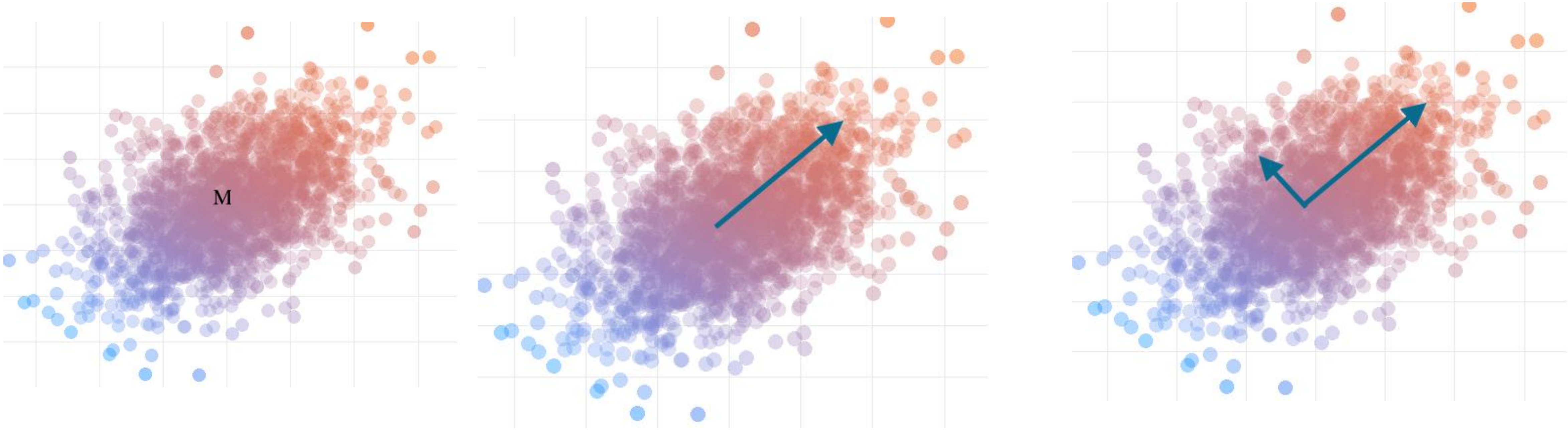
Curse of Dimensionality



Curse of Dimensionality



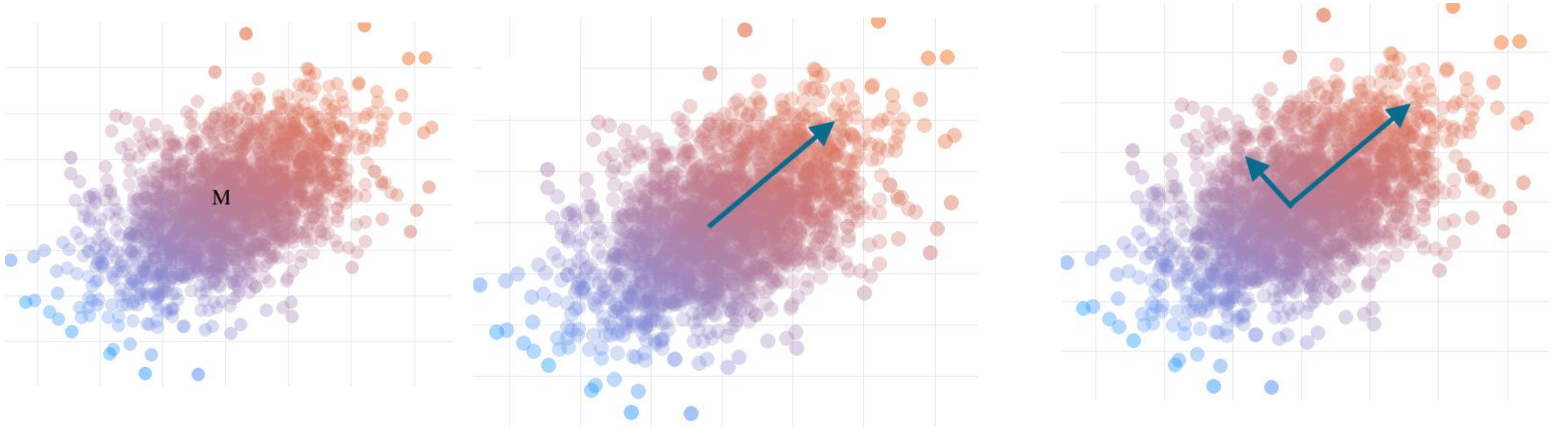
Dimensionality Reduction - PCA



Dimensionality Reduction - PCA

Data $D \in \mathbb{R}^{m \times n}$:
m datapoints in n dimensions

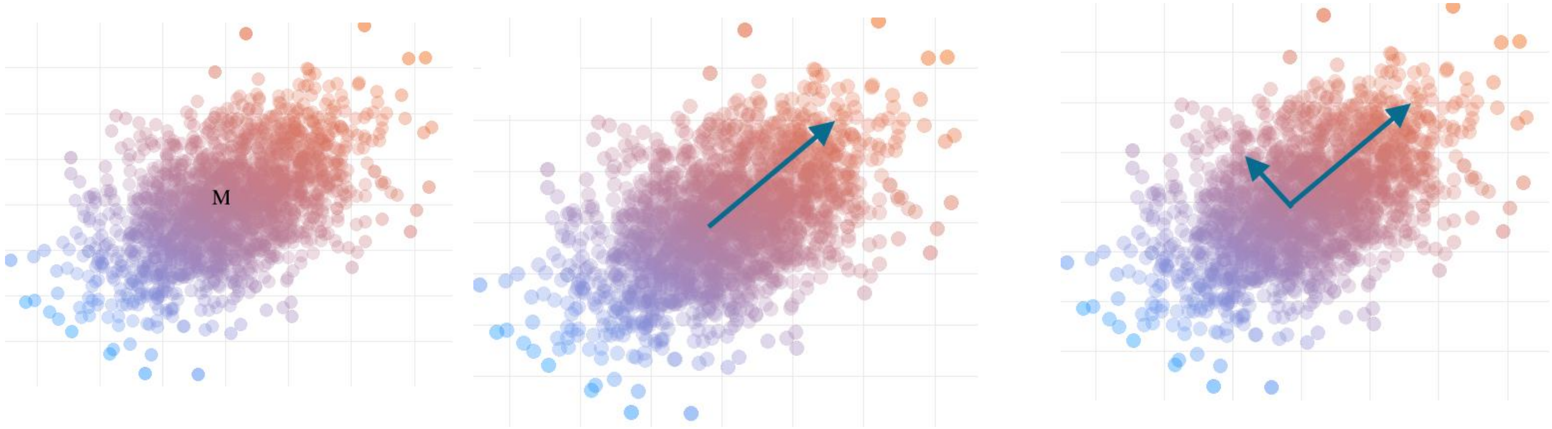
Low-rank data $U \in \mathbb{R}^{m \times n}$:
m datapoints spanning d dimensions



Dimensionality Reduction - PCA

Data $D \in \mathbb{R}^{m \times n}$:
m datapoints in n dimensions

Low-rank data $U \in \mathbb{R}^{m \times n}$:
m datapoints spanning d dimensions



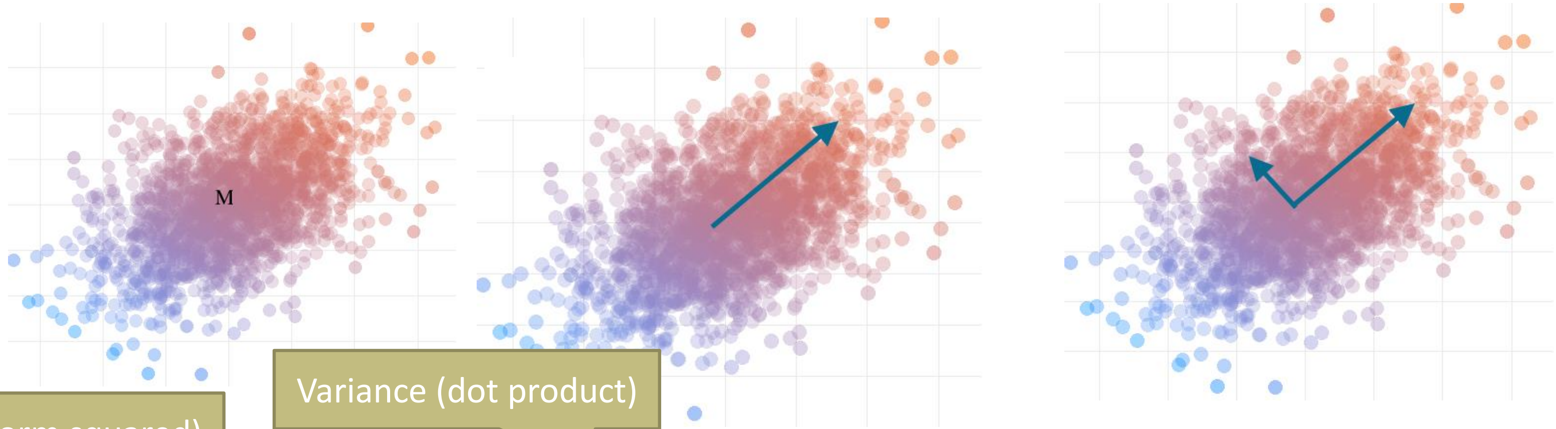
$$\min_{U:\mathrm{rank}(U)=d} \|D - U\|_F^2 = \min_{P \in \mathcal{P}_d} \|D - DP\|_F^2 = |D|_F^2 - \max_{P \in \mathcal{P}_d} D^T D \cdot P$$

$$\mathcal{P}_d = \{P \in \mathbb{R}^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

Dimensionality Reduction - PCA

Data $D \in \mathbb{R}^{m \times n}$:
m datapoints in n dimensions

Low-rank data $U \in \mathbb{R}^{m \times n}$:
m datapoints spanning d dimensions



Reconstruction error (norm squared)

Variance (dot product)

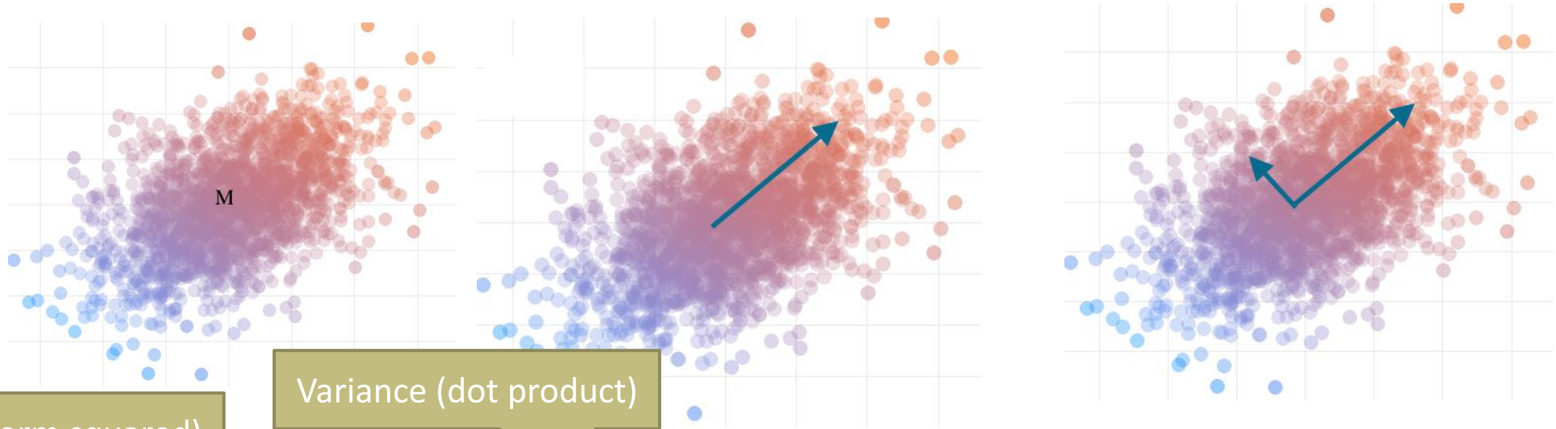
$$\min_{U:\mathrm{rank}(U)=d} \|D - U\|_F^2 = \min_{P \in \mathcal{P}_d} \|D - DP\|_F^2 = |D|_F^2 - \max_{P \in \mathcal{P}_d} D^T D \cdot P$$

$$\mathcal{P}_d = \{P \in \mathbb{R}^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

Dimensionality Reduction - PCA

Data $D \in \mathbb{R}^{m\times n}$:
m datapoints in n dimensions

Low-rank data $U \in \mathbb{R}^{m\times n}$:
m datapoints spanning d dimensions



Reconstruction error (norm squared)

Variance (dot product)

$$\min_{U:\mathrm{rank}(U)=d} \|D-U\|_F^2 = \min_{P\in\mathcal{P}_d}\|D-DP\|_F^2 = |D|_F^2 - \max_{P\in\mathcal{P}_d} D^T D\cdot P$$

$$\mathcal{P}_d = \{P \in \mathbb{R}^{n\times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

Easily solved ($O(n^3)$ time) by SVD (Singular Value Decomposition):

- $U = DLL^{\mathsf{T}}$ for some orthonormal $L \in \mathbb{R}^{n\times d}$. (Hence the 1st equality above)

- Columns of L are the top d-singular vectors of D.

Unfair PCA

Reconstruction error (norm squared)

Variance (dot product)

$$\min_{U:\text{rank}(U)=d} \|D - U\|_F^2 = \min_{P \in \mathcal{P}_d} \|D - DP\|_F^2 = |D|_F^2 - \max_{P \in \mathcal{P}_d} D^T D \cdot P$$

$$\mathcal{P}_d = \{P \in \mathbb{R}^{n \times n} : P\ symmetric, rank(P) = d, P^2 = P\}$$

Unfair PCA

Reconstruction error (norm squared)

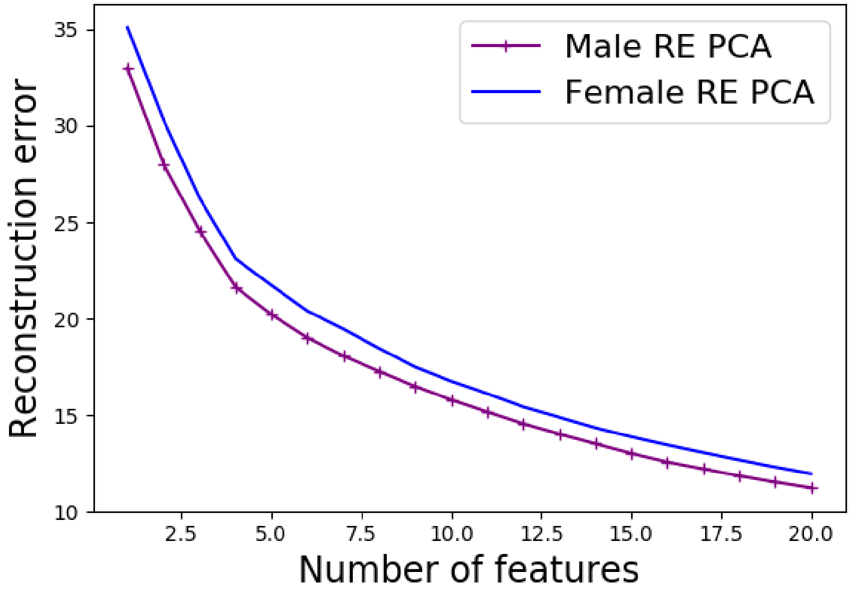
Variance (dot product)

$$\min_{U:\mathrm{rank}(U)=d} \|D - U\|_F^2 = \min_{P\in\mathcal{P}_d} \|D - DP\|_F^2 = |D|_F^2 - \max_{P\in\mathcal{P}_d} D^T D \cdot P$$

$$\mathcal{P}_d = \{P \in \mathbb{R}^{n\times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

Standard PCA on face data LFW of male and female.

Average reconstruction error (RE) of PCA on LFW



Unfair PCA

Reconstruction error (norm squared)

Variance (dot product)

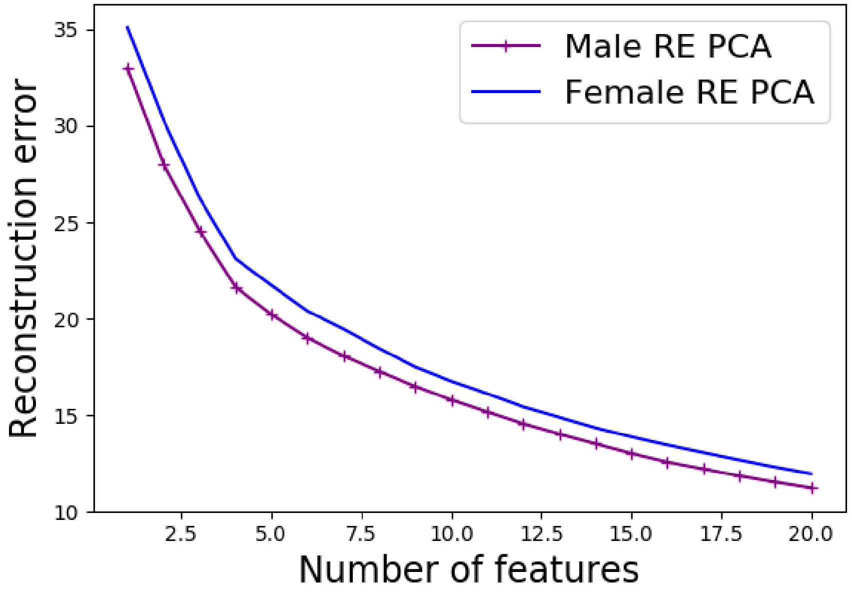
$$\min_{U:\mathrm{rank}(U)=d} \|D - U\|_F^2 = \min_{P \in \mathcal{P}_d} \|D - DP\|_F^2 = |D|_F^2 - \max_{P \in \mathcal{P}_d} D^T D \cdot P$$

$$\mathcal{P}_d = \{P \in \mathbb{R}^{n \times n} : P \text{ symmetric}, \mathrm{rank}(P) = d, P^2 = P\}$$

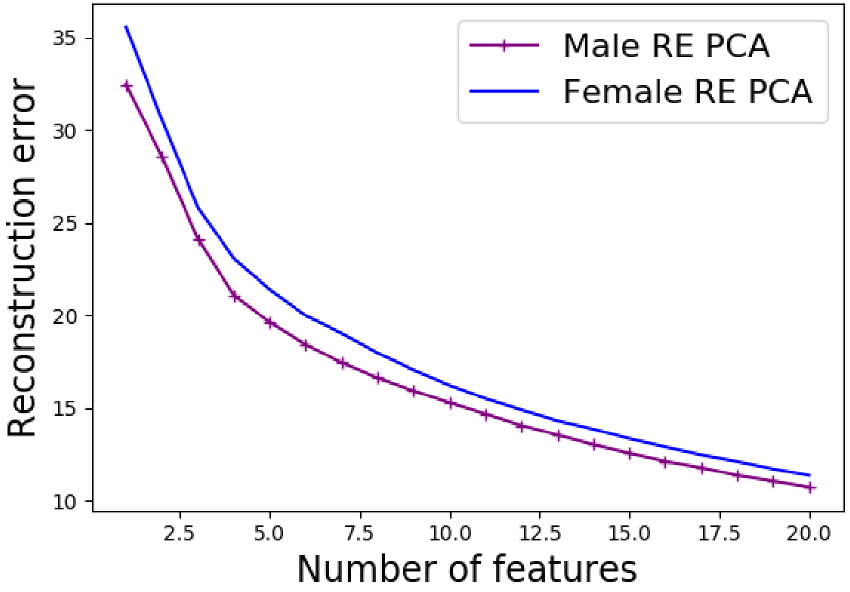
Standard PCA on face data LFW of male and female.

Equalizing male and female weight before PCA

Average reconstruction error (RE) of PCA on LFW



Average reconstruction error (RE) of PCA on LFW (resampled)



Fair PCA

- Given data matrices $D_i \in \mathbb{R}^{m_i \times n}$ for $i = 1, \ldots, k$ and a projection matrix $P \in \mathbb{R}^{n \times n}$.
- $Err(D_i, P) = |D_i - D_i P|_F^2 = Tr\left(D_i^T D_i\right) - D_i^T D_i \cdot P$
- Given target dimension $d < n$.
- Fair PCA's task: Find a projection matrix P of rank at most d that minimizes the maximum error.

Fair PCA:= $\min_{P \in \mathcal{P}_d} \max_{i \in [k]} Err(D_i, P)$

$\mathcal{P}_d = \{P \in R^{n \times n} : P \ symmetric, rank(P) = d, P^2 = P\}$

Fair PCA as rank constrained SDP:

$\min z$ s.t.

$$z \geq Tr\left(D_i^T D_i\right) - D_i^T D_i \cdot P \quad \forall i = 1, \ldots, k$$
$$rank(P) = d$$
$$0 \preccurlyeq P \preccurlyeq I$$

Fair PCA

- Given data matrices $D_i \in \mathbb{R}^{m_i \times n}$ for $i = 1, \ldots, k$ and a projection matrix $P \in \mathbb{R}^{n \times n}$.
- $Err(D_i, P) = |D_i - D_i P|_F^2 = Tr\left(D_i^T D_i\right) - D_i^T D_i \cdot P$
- Given target dimension $d < n$.
- Fair PCA's task: Find a projection matrix P of rank at most d that minimizes the maximum error.

Fair PCA:= $\min_{P \in \mathcal{P}_d} \max_{i \in [k]} Err(D_i, P)$

$\mathcal{P}_d = \{P \in R^{n \times n} : P\ symmetric, rank(P) = d, P^2 = P\}$

Fair PCA as rank constrained SDP:

min z s.t.

$$z \geq Tr\left(D_i^T D_i\right) - D_i^T D_i \cdot P \quad \forall i = 1, \ldots, k$$

$$rank(P) = d$$

$$0 \preccurlyeq P \preccurlyeq I$$

- Note: standard SVD won't work

Fair Dimensionality Reduction

- More generally, we are given utility functions $u_i: \mathcal{P}_d \to \mathbb{R}$ that measure the utility of each group.
- Moreover, we are given a function $g: \mathbb{R}^k \to \mathbb{R}$ that combines these utilities to the "social utility".

Fair DR:= $\max_{P \in \mathcal{P}_d} g\big(u_1(P), u_2(P), \ldots, u_k(P)\big)$

$\mathcal{P}_d = \{P \in R^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$

Fair PCA:= $\min_{P \in \mathcal{P}_d} \max_{i \in [k]} Err(D_i, P)$

$\mathcal{P}_d = \{P \in R^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$

Fair PCA: Special case with $u_i(P) = -Err(D_i, P)$ and $g(.) = \min$

$Loss_i(P) = \|D_i - D_i P\|_2^2 - \left\|D_i - D_i P_i^*\right\|_2^2$

where P_i^* is the best rank d projection for group i.

- Loss for being part of the other groups.

Fair Dimensionality Reduction

- More generally, we are given utility functions $u_i: \mathcal{P}_d \to \mathbb{R}$ that measure the utility of each group.
- Moreover, we are given a function $g: \mathbb{R}^k \to \mathbb{R}$ that combines these utilities to the “social utility”.

Fair DR:=$\max_{P \in \mathcal{P}_d} g\big(u_1(P), u_2(P), \dots, u_k(P)\big)$

$$\mathcal{P}_d = \{P \in R^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

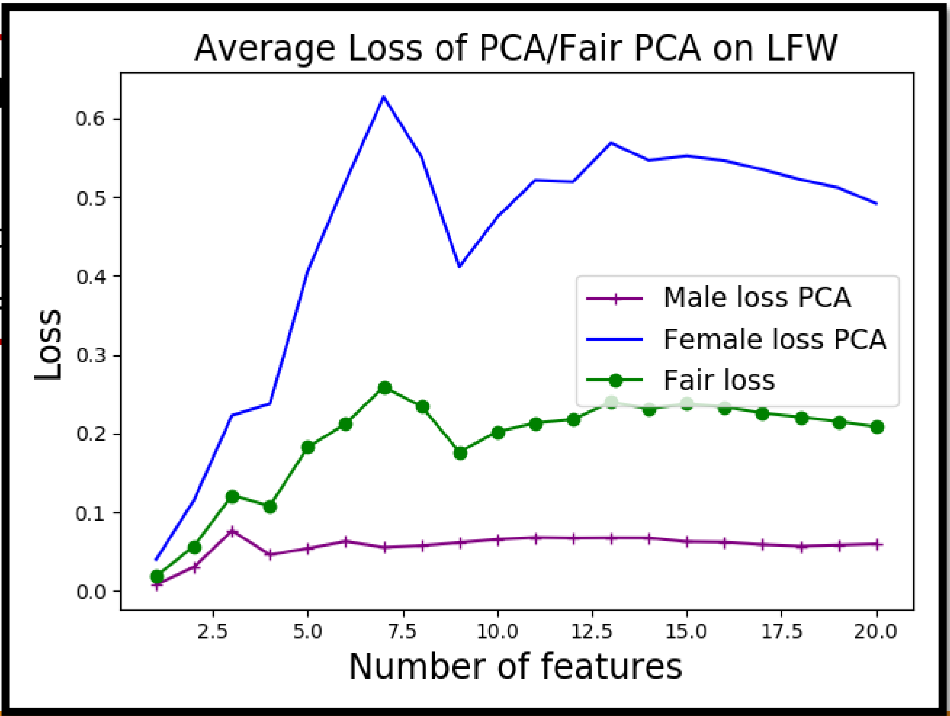
Fai

Fair PCA: Special case with $u_i(P) = -Err(D_i, P)$ and $g(.) = \min$

$$Loss_i(P) = \|D_i - D_i P\|_2^2 - \left\|D_i - D_i P_i^*\right\|_2^2$$

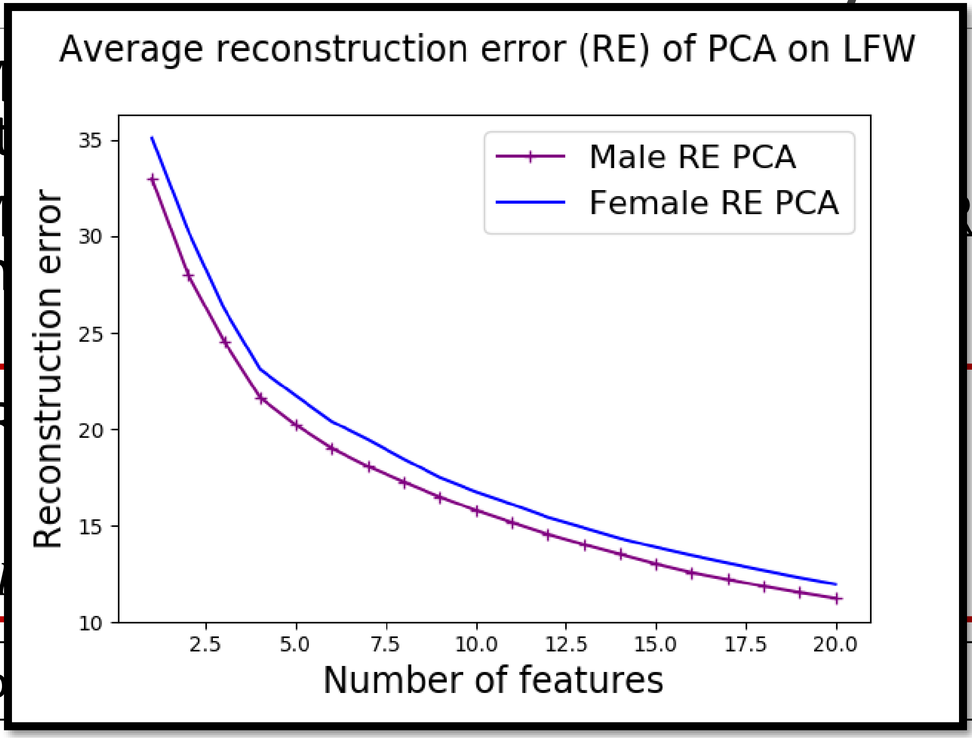
where P_i^* is the best rank d projection for group i.

- Loss for being part of the other groups.



Fair Dimensionality Reduction

- M… ctions $u_i: \mathcal{P}_d \to \mathbb{R}$ that measure the ut…
- M… $^k \to \mathbb{R}$ that combines these utilities to th…



Fair DR…

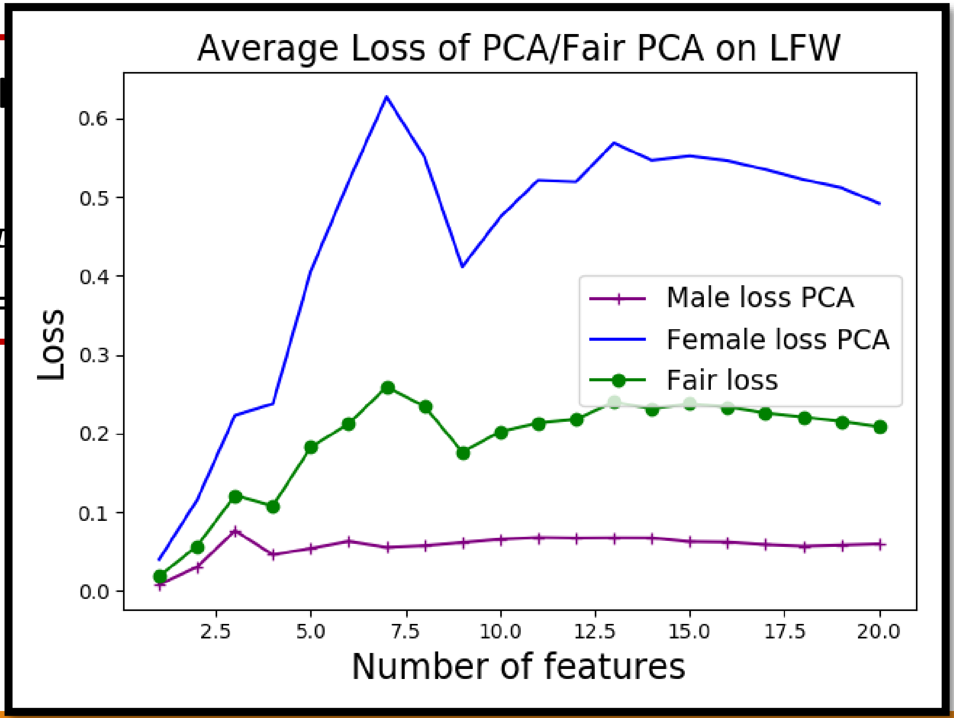
$\mathcal{P}_d = \{$…

Fai…

$\mathcal{F}$…

=

Fair PCA: Sp… min



$Loss_i(P) = \|D_i - D_i P\|_2^2 - \left\|D_i - D_i P_i^*\right\|_2^2$

where P_i^* is the best rank d projection for group i.

- Loss for being part of the other groups.

Related work to fair DM formulation

- Rank Constrained SDPs are widely used.
  - Signal processing[Davies and Eldar’12, Ahmed and Romberg’15]
  - Distance Matrices: Localization sensors [So and Ye’07], nuclear magnetic resonance spectroscopy [Singer’08]
  - Item Response Data, Recommendation Systems[Goldberg et al’93]
  - Machine Learning: Multi-task Learning [Obozinski, Taskar, Jordan’10], Natural Language Processing[Blei’12]
  - Survey by [Davenport, Romberg’2016]
- Work by Barvinok’95, Pataki’98 on characterizations of extreme points of SDPs.
  - Algorithmic work by [Burer, Monteiro’03].
  - Related to S-Lemma [Yakubovich’71].

The problem

- Motivation → Define fair PCA and fair dimensional reduction (fair DM)

The method

- Two of our algorithms (focus on one) and their analysis

Conclusion

Outline

Main result

Fair PCA:= $\min_{P \in \mathcal{P}_d} \max_{i \in [k]} Err(D_i, P) := |D_i - D_i P|_F^2$

$$\mathcal{P}_d = \{P \in R^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

- There is a polynomial time algorithm for the Fair PCA problem that returns a rank at most $d + \sqrt{2k + \frac{1}{4}} - \frac{3}{2}$ whose objective is better than the optimum.
  - 2 groups → solved exactly

Main result

Fair PCA:= $\min_{P\in\mathcal{P}_d} \max_{i\in[k]} Err(D_i, P) \coloneqq |D_i - D_i P|_F^2$

$$\mathcal{P}_d = \{P \in R^{n\times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

- There is a polynomial time algorithm for the Fair PCA problem that returns a rank at most $d + \sqrt{2k + \frac{1}{4}} - \frac{3}{2}$ whose objective is better than the optimum.
  - 2 groups → solved exactly
- Also generalize to fair DM when u_i is linear and g is concave

Fair DR:= $\max_{P\in\mathcal{P}_d} \ g\left(u_1(P), u_2(P), \ldots, u_k(P)\right)$

$$\mathcal{P}_d = \{P \in R^{n\times n}: P\ symmetric, rank(P) = d, P^2 = P\}$$

Main result

Fair PCA:= $\min_{P \in \mathcal{P}_d} \max_{i \in [k]} Err(D_i, P) := |D_i - D_i P|_F^2$

$\mathcal{P}_d = \{P \in R^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$

- There is a polynomial time algorithm for the Fair PCA problem that returns a rank at most $d + \sqrt{2k + \frac{1}{4}} - \frac{3}{2}$ whose objective is better than the optimum.
  - 2 groups → solved exactly
- Also generalize to fair DM when u_i is linear and g is concave

Fair DR:= $\max_{P \in \mathcal{P}_d} g(u_1(P), u_2(P), \ldots, u_k(P))$

$\mathcal{P}_d = \{P \in R^{n \times n}: P\ symmetric, rank(P) = d, P^2 = P\}$

- Note: convertible to approximation ratio guarantee (no rank violation)

SDP relaxation

Rank-Constrained SDP

$$\min C \cdot X$$
$$A_i \cdot X \leq b_i \; \forall i = 1, \ldots, m$$
$$rank(X) \leq d$$
$$0 \preccurlyeq X \preccurlyeq I$$

SDP-Relaxation

$$\min C \cdot X$$
$$A_i \cdot X \leq b_i \; \forall i = 1, \ldots, m$$
$$trace(X) \leq d$$
$$0 \preccurlyeq X \preccurlyeq I$$

Fair PCA as rank constrained SDP:

$\min z$ s.t.

$$z \geq Tr\left(D_i^T D_i\right) - D_i^T D_i \cdot P \quad \forall i = 1, \ldots, k$$
$$rank(P) = d$$
$$0 \preccurlyeq P \preccurlyeq I$$

SDP extreme points

Rank-Constrained SDP

$$\min C \cdot X$$
$$A_i \cdot X \le b_i \; \forall i = 1, \dots, m$$
$$rank(X) \le d$$
$$0 \preccurlyeq X \preccurlyeq I$$

SDP-Relaxation

$$\min C \cdot X$$
$$A_i \cdot X \le b_i \; \forall i = 1, \dots, m$$
$$trace(X) \le d$$
$$0 \preccurlyeq X \preccurlyeq I$$

Our key theorem: Every extreme point of the SDP-Relaxation has rank at most $d + \sqrt{2m + 9/4} - 3/2$.

Our main result follows: There is a polynomial time algorithm for the Fair PCA problem that returns a rank at most $\mathrm{d} + \sqrt{2k + \frac{1}{4}} - \frac{3}{2}$ whose objective is better than the optimum.

Proof

$$\min C \cdot X$$
$$A_i \cdot X \leq b_i \ \ \forall i = 1, \ldots, m.$$
$$Tr(X) \leq d$$
$$0 \preccurlyeq X \preccurlyeq I$$

Theorem 2: Every extreme point of the SDP-Relaxation has rank at most $d + \sqrt{2m + 9/4} - 3/2$.

Proof: Let X be an extreme point with r fractional eigenvalues.

$$X = [U_1 \ U_f \ U_0] \ [diag(1) \ D \ 0][U_1 \ U_f \ U_0]^{\mathrm{T}} = U_1 U_1^T + U_f D U_f^T$$

D is $r \times r$ diagonal matrix with $0 < D_{ii} < 1$ and $[U_1 \ U_f \ U_0]$ is a orthogonal matrix of eigenvectors.

Claim: If $\frac{r(r+1)}{2} > m + 1$ then there exists a $r \times r$ symmetric matrix $F \neq 0$ such that

$Y = U_1 U_1^T + U_f (D + F) U_f^T$ and $Z = U_1 U_1^T + U_f (D - F) U_f^T$ are feasible.

Assuming the claim, we get a contradiction to the definition of the extreme point.

The proof is then finished ($r \leq \sqrt{2m + 9/4} - 3/2$).

Fact: Eigenvalues of Y are same as eigenvalues of $[diag(1); \ D + F; \ 0]$ and eigenvalues of Z are same as eigenvalues of $[diag(1); \ D - F; \ 0]$ (useful for checking $0 \preccurlyeq X \preccurlyeq I$ condition).

Claim: If $\frac{r(r+1)}{2} > m+1$ then there exists a $r \times r$ symmetric matrix $F \neq 0$ such that

$U_1 U_1^T + U_f(D+F)U_f^T$ and $U_1 U_1^T + U_f(D-F)U_f^T$ are feasible.

$$\min C \cdot X$$
$$A_i \cdot X \leq b_i \;\; \forall i = 1, \dots, m.$$
$$Tr(X) \leq d$$
$$0 \preccurlyeq X \preccurlyeq I$$

Proof: Consider the linear system:

$$A_i \cdot U_f G U_f^T = 0 \; \forall i = 1, \dots, m.$$
$$Tr\left(U_f G U_f^T\right) = 0$$
$$G_{ij} = G_{ji} \quad \forall i \neq j$$

- Number of equations $m + 1 + \frac{r(r-1)}{2}$. Number of variables r^2.
- If $r^2 > m + 1 + \frac{r(r-1)}{2}$ (more freedom than constraints), then there is a line of solutions, i.e. , $G \neq 0$ such that $\{\lambda\, G : \lambda \in R\}$ all satisfy the above constraints.
- Consider $F = \epsilon G$ for small enough $\epsilon > 0$.

Check feasibility:

- $U_1 U_1^T + U_f(D \pm F)U_f^T$ keeps the same dot product and trace.
- Eigenvalues of $[diag(1);\; D+F;\; 0]$ = $[diag(1);\; D+\epsilon G;\; 0]$ remain bounded away from 0 and 1

SDP extreme points summary

Theorem: Every extreme point of the SDP

$$\min C \cdot X$$

$$A_i \cdot X \leq b_i \quad \forall i = 1, \dots, m$$

$$Tr(X) \leq d$$

$$0 \preccurlyeq X \preccurlyeq I$$

has rank at most $d + \sqrt{2m + 9/4} - \frac{3}{2}$.

Generalizes Barvinok'95, Pataki'98 (similar result, without $\preccurlyeq I$ and trace constraints)

Iterative Rounding

Theorem: There is an iterative rounding algorithm that given

$$\begin{aligned} &\min C \cdot X \\ &A_i \cdot X \leq b_i \qquad \forall i = 1, \ldots, m \\ &Tr(X) \leq d \\ &0 \preccurlyeq X \preccurlyeq I \end{aligned}$$

with optimal solution X^* returns a feasible solution Y s.t.

1. $\mathrm{rank}(Y) \leq d$.
2. $C \cdot Y \leq C \cdot X^*$.
3. $A_i \cdot Y \geq A_i \cdot X^* - \Delta$

Where $\Delta = \max_{S \subseteq [m]} \sum_{i=1}^{\sqrt{2|S|}} \sigma_i \left(\frac{1}{|S|} \sum_{j \in S} A_j \right)$ where $\sigma_i(B)$ is the i^{th} largest singular value of B.

[More details in the paper]

Idea: Fix eigenvalues to 0 and 1.

Fix two subspaces F_0 and F_1 for corresponding eigenfaces.

Update SDP to work only in the orthogonal space F.

Show a constraint can be removed or one of the eigenvalues is 0 or 1.

Other results

- The Fair PCA problem is polynomial time solvable for constant k and d.
  - Algorithmic theory of quadratic maps. [Grigoriev and Pasechnik ’05]

- Problem is NP-hard for general k, d=1.

- Experiments
  - SDP relaxation performs optimally (exact rank) almost always in practice
  - Runtime of SDP works up to $n \approx 75$ but alternative (multiplicative weight) works in practice equally well up to $n \approx 1500$
  - All publicly available (Github)

Conclusion

- Formulating the fairness in dimensionality reduction
- Propose two new algorithms (SDP extreme points and SDP rounding)
- Their analysis uses low-rank property of SDP extreme point and by itself maybe of interest for optimization community
- Open question: more application of low-rank properties?

Thanks!