Fair Clustering Through Fairlets

(EE531 Final Project - Fairness)

F. Chierichetti[1] R. Kumar[2] S. Lattanzi[2] S. Vassilvitskii[2]

[1]Dipartimento di Informatica, Sapienza University
[2]Google Research

Appeared at NIPS 2017

Table of Contents

Outline

Disparate Impact

- **Griggs v. Duke Power Co.**[1]: the most significant U.S.S.C. rulings on disparate(or adverse) impact.

[1] *Griggs v. Duke Power Co., 401 U.S. 424 (Supreme Court of the United States 1971).*

[2] *Uniform Guidelines on Employee Selection Procedures.*

[3] Dan Biddle. *Adverse Impact and Test Validation: A Practitioner's Guide to Valid and Defensible Employment Testing.* 2nd ed. Routledge, 2006.

Disparate Impact

- **Griggs v. Duke Power Co.**[1]: the most significant U.S.S.C. rulings on disparate(or adverse) impact.
- "Congress has now provided that tests or criteria for employment or promotion may-not provide equality of opportunity merely in the sense of the fabled offer of milk to the stork and the fox"

[1] *Griggs v. Duke Power Co., 401 U.S. 424 (Supreme Court of the United States 1971).*

[2] *Uniform Guidelines on Employee Selection Procedures.*

[3] Dan Biddle. *Adverse Impact and Test Validation: A Practitioner's Guide to Valid and Defensible Employment Testing.* 2nd ed. Routledge, 2006.

Disparate Impact

- **Griggs v. Duke Power Co.**[1]: the most significant U.S.S.C. rulings on disparate(or adverse) impact.
- "Congress has now provided that tests or criteria for employment or promotion may-not provide equality of opportunity merely in the sense of the fabled offer of milk to the stork and the fox"
- Disparate impact: "substantially different rate of selection in hiring, promotion, or other employment decision which works to the disadvantage of members of a race, sex, or ethnic group"[2]

[1] *Griggs v. Duke Power Co., 401 U.S. 424 (Supreme Court of the United States 1971).*

[2] *Uniform Guidelines on Employee Selection Procedures.*

[3] Dan Biddle. *Adverse Impact and Test Validation: A Practitioner's Guide to Valid and Defensible Employment Testing.* 2nd ed. Routledge, 2006.

Disparate Impact

- **Griggs v. Duke Power Co.**[1]: the most significant U.S.S.C. rulings on disparate(or adverse) impact.
- "Congress has now provided that tests or criteria for employment or promotion may-not provide equality of opportunity merely in the sense of the fabled offer of milk to the stork and the fox"
- Disparate impact: "substantially different rate of selection in hiring, promotion, or other employment decision which works to the disadvantage of members of a race, sex, or ethnic group"[2]
- cf. 80%-rule[3]

---

[1] *Griggs v. Duke Power Co., 401 U.S. 424 (Supreme Court of the United States 1971).*

[2] *Uniform Guidelines on Employee Selection Procedures.*

[3] Dan Biddle. *Adverse Impact and Test Validation: A Practitioner's Guide to Valid and Defensible Employment Testing.* 2nd ed. Routledge, 2006.

Introduction

- In this work, the notion of disparate impact, as described previously, will be followed!

Introduction

- In this work, the notion of disparate impact, as described previously, will be followed!
- Question: How can we formalize this notion of disparate impact in the case of clustering problem?

Outline

Previous / Related Works

- Two "big" tracks in fairness research:
  - Codifying the meaning of fairness in algorithms
  - Modifying algorithms to make it achieve fair outcomes under a specific notion of fairness

[4] Michael Feldman et al. "Certifying and Removing Disparate Impact". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*. 2015, pp. 259–268.

Previous / Related Works

- Two "big" tracks in fairness research:
  - Codifying the meaning of fairness in algorithms
  - Modifying algorithms to make it achieve fair outcomes under a specific notion of fairness
- In the case of disparate impact, Feldman *et al.*[4] did some work in the first track.

[4] Michael Feldman et al. "Certifying and Removing Disparate Impact". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*. 2015, pp. 259–268.

- Two "big" tracks in fairness research:
  - Codifying the meaning of fairness in algorithms
  - Modifying algorithms to make it achieve fair outcomes under a specific notion of fairness
- In the case of disparate impact, Feldman *et al.*[4] did some work in the first track.
- This work is similar to the second track, but one of the first in the unsupervised learning tasks.

[4] Michael Feldman et al. "Certifying and Removing Disparate Impact". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*. 2015, pp. 259–268.

Previous / Related Works

- Two "big" tracks in fairness research:
  - Codifying the meaning of fairness in algorithms
  - Modifying algorithms to make it achieve fair outcomes under a specific notion of fairness
- In the case of disparate impact, Feldman *et al.*[4] did some work in the first track.
- This work is similar to the second track, but one of the first in the unsupervised learning tasks.
- Unlike other works, *strong guarantees* on the quality of any fair clustering solution.

[4] Michael Feldman et al. "Certifying and Removing Disparate Impact". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*. 2015, pp. 259–268.

Previous / Related Works

- The general framework of this work follows that of Zemel *et al.*[5].

[5] Richard S. Zemel et al. "Learning Fair Representations". In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. 2013, pp. 325–333.

Previous / Related Works

- The general framework of this work follows that of Zemel *et al.*[5].
- Learn a set of intermediate representations to satisfy two competing goals:

[5] Richard S. Zemel et al. "Learning Fair Representations". In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. 2013, pp. 325–333.

Previous / Related Works

- The general framework of this work follows that of Zemel *et al.*[5].
- Learn a set of intermediate representations to satisfy two competing goals:
  - The intermediate representation should encode the data as well as possible.
  - The encoded representation is sanitized in the sense that it should be blind to whether or not the individual is from the protected group.

[5] Richard S. Zemel et al. "Learning Fair Representations". In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. 2013, pp. 325–333.

Previous / Related Works

- The general framework of this work follows that of Zemel *et al.*[5].
- Learn a set of intermediate representations to satisfy two competing goals:
  - The intermediate representation should encode the data as well as possible.
  - The encoded representation is sanitized in the sense that it should be blind to whether or not the individual is from the protected group.
- Using this, any classification algorithm can be transformed into a fair classifier, by simply *applying the classifer to the sanitized representation of the data*.

[5] Richard S. Zemel et al. "Learning Fair Representations". In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. 2013, pp. 325–333.

Previous / Related Works

- This work is also closely related to that of Zafar *et al.*[6].

[6] Muhammad Bilal Zafar et al. "Fairness Constraints: Mechanisms for Fair Classification". In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*. 2017, pp. 962–970.

Previous / Related Works

- This work is also closely related to that of Zafar *et al.*[6].
- Part of their work was focused on designing a convex margin-based classifier that maximizes accuracy subject to fairness constraints, and helps ensure compliance with a non-discrimination policy or law (e.g., a given p%-rule)

[6] Muhammad Bilal Zafar et al. "Fairness Constraints: Mechanisms for Fair Classification". In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*. 2017, pp. 962–970.

Previous / Related Works

- This work is also closely related to that of Zafar *et al.*[6].
- Part of their work was focused on designing a convex margin-based classifier that maximizes accuracy subject to fairness constraints, and helps ensure compliance with a non-discrimination policy or law (e.g., a given p%-rule)
- This work addresses an open question in that work, which asked for a general framework to solve an unsupervised learning task respecting the p%-rule.

[6] Muhammad Bilal Zafar et al. "Fairness Constraints: Mechanisms for Fair Classification". In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*. 2017, pp. 962–970.

Outline

k-clustering

Definition

Let (M, d) be a metric space, equipped with the metric function d. Given a set of points $X \subset M$, a *k-clustering* of X is a partition of X into k disjoint subsets, $C_1, \ldots, C_k$, called *clusters*.

k-clustering

Definition

Let (M, d) be a metric space, equipped with the metric function d. Given a set of points $X \subset M$, a *k-clustering* of X is a partition of X into k disjoint subsets, $C_1, \ldots, C_k$, called *clusters*.

Alternate Formulation

A *k-clustering* of X is an *assignment function*, $\alpha : X \to [k]$.
Each cluster C_i is the preimage of i under α i.e. $C_i = \alpha^{-1}(i)$

k-clustering

Definition

Let (M, d) be a metric space, equipped with the metric function d. Given a set of points $X \subset M$, a *k-clustering* of X is a partition of X into k disjoint subsets, $C_1, \ldots, C_k$, called *clusters*.

Alternate Formulation

A *k-clustering* of X is an *assignment function*, $\alpha : X \to [k]$.
Each cluster C_i is the preimage of i under α i.e. $C_i = \alpha^{-1}(i)$

- There are many ways to quantify "how good a given clustering is"
- Depending on the objective, different variants of clustering problems are possible.
- Here, we consider two specific types of k-clustering.

k-center problem

Problem

Given a set of points $X \subset M$, find a k-clustering of X, denoted as $\mathcal{C}$, that minimizes

$$\phi(X, \mathcal{C}) = \max_{C \in \mathcal{C}} \left[\min_{c \in C} \max_{x \in C} d(x, c) \right]$$

k-median problem

Problem

Given a set of points $X \subset M$, find a k-clustering of X, denoted as $\mathcal{C}$, that minimizes

$$\psi(X, \mathcal{C}) = \sum_{C \in \mathcal{C}} \left[\min_{c \in C} \sum_{x \in C} d(x, c) \right]$$

Fair clustering?

- In order to consider a "fair" version of clustering, we first have to identify the *unprotected attribute* and *protected attribute*

Fair clustering?

- In order to consider a "fair" version of clustering, we first have to identify the *unprotected attribute* and *protected attribute*
- We shall consider the *coordinate* as the unprotected attribute.

Fair clustering?

- In order to consider a "fair" version of clustering, we first have to identify the *unprotected attribute* and *protected attribute*
- We shall consider the *coordinate* as the unprotected attribute.
- For simplicity, let us represent the protected attribute as the *coloring* of the points.

Fair clustering?

- In order to consider a "fair" version of clustering, we first have to identify the *unprotected attribute* and *protected attribute*
- We shall consider the *coordinate* as the unprotected attribute.
- For simplicity, let us represent the protected attribute as the *coloring* of the points.
- To simplify things further (as in the paper), let us only consider the case of binary coloring.

Fair clustering?

For $Y \subset X$, let us denote:

- $\chi : X \to \{\text{RED}, \text{BLUE}\}$ is the given binary coloring.
- $R(Y) = \{x \in X : \chi(x) = \text{RED}\}$, $r(Y) = |R(Y)|$
- $B(Y) = \{x \in X : \chi(x) = \text{BLUE}\}$, $b(Y) = |B(Y)|$

Fair clustering?

For $Y \subset X$, let us denote:

- $\chi : X \rightarrow \{\text{RED}, \text{BLUE}\}$ is the given binary coloring.
- $R(Y) = \{x \in X : \chi(x) = \text{RED}\}$, $r(Y) = |R(Y)|$
- $B(Y) = \{x \in X : \chi(x) = \text{BLUE}\}$, $b(Y) = |B(Y)|$

Definition

For $\emptyset \neq Y \subset X$, the *balance* of Y is defined as:

$$\text{balance}(Y) = \min\left(\frac{r(Y)}{b(Y)}, \frac{b(Y)}{r(Y)}\right) \in [0, 1]$$

The *balance* of a clustering $\mathcal{C}$ is defined as:

$$\text{balance}(\mathcal{C}) = \min_{C \in \mathcal{C}} \text{balance}(C)$$

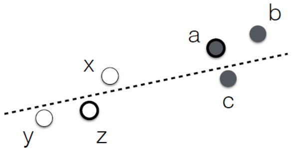
- If balance(Y) is 0(resp. 1), Y is fully unbalanced(resp. perfectly balanced)

Fair clustering?

- A clustering algorithm is *colorblind* if it doesn't take the protected attribute (coloring) into its decision making.

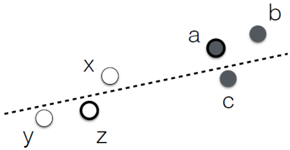
Fair clustering?

- A clustering algorithm is *colorblind* if it doesn't take the protected attribute (coloring) into its decision making.
- Colorblind algorithm may result in a very unfair clustering. (Unfair in the sense that the resulting clustering is very unbalanced)



Fair clustering?

- A clustering algorithm is *colorblind* if it doesn't take the protected attribute (coloring) into its decision making.
- Colorblind algorithm may result in a very unfair clustering. (Unfair in the sense that the resulting clustering is very unbalanced)



- Therefore a "fair" clustering must take into account not just the position of the centers, but also the assignment function!

Balance

Lemma 2(Combination)

Let $Y, Y' \subset X$ be disjoint.
If $\mathcal{C}$ and $\mathcal{C}'$ are clusterings of Y and Y', respectively, then

$$\text{balance}(\mathcal{C} \cup \mathcal{C}') = \min(\text{balance}(\mathcal{C}), \text{balance}(\mathcal{C}'))$$

- For any clustering $\mathcal{C}$ of X, we have $\text{balance}(\mathcal{C}) \leq \text{balance}(X)$.
- If X is not perfectly balanced, then no clustering of X can be perfectly balanced.

Toward fairlets

Definition

Let b, r be some integers such that $1 \leq b \leq r$ and $\gcd(b, r) = 1$.

- A clustering $\mathcal{Y}$ of X is called a $(b, r)-$*fairlet decomposition of* X if (i) $\forall Y \in \mathcal{Y}\ |Y| \leq b + r$ and (ii) $\text{balance}(\mathcal{Y}) = b/r = \text{balance}(X)$
- Each $Y \in \mathcal{Y}$ is called a $(b, r)-$*fairlet*, or simply *fairlet*.

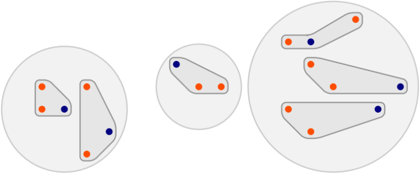
Toward fairlets

Definition

Let b, r be some integers such that $1 \leq b \leq r$ and $\gcd(b, r) = 1$.

- A clustering $\mathcal{Y}$ of X is called a $(b, r)-$*fairlet decomposition of* X if (i) $\forall Y \in \mathcal{Y}\ |Y| \leq b + r$ and (ii) $\text{balance}(\mathcal{Y}) = b/r = \text{balance}(X)$
- Each $Y \in \mathcal{Y}$ is called a $(b, r)-$*fairlet*, or simply *fairlet*.

- Fairlet can be thought of as a group of points that are fair and cannot be split further into true subsets that are also fair.



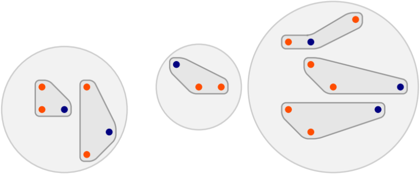
Toward fairlets

Definition

Let b, r be some integers such that $1 \leq b \leq r$ and $\gcd(b, r) = 1$.

- A clustering $\mathcal{Y}$ of X is called a $(b, r)-$*fairlet decomposition of* X if (i) $\forall Y \in \mathcal{Y}\ |Y| \leq b + r$ and (ii) $\text{balance}(\mathcal{Y}) = b/r = \text{balance}(X)$
- Each $Y \in \mathcal{Y}$ is called a $(b, r)-$*fairlet*, or simply *fairlet*.

- Fairlet can be thought of as a group of points that are fair and cannot be split further into true subsets that are also fair.



- Intuitively, the balance of the original set of points is preserved while keeping each cluster "small".

Toward fairlets

Lemma 3

Let $\text{balance}(X) = b/r$ for some integers $1 \leq b \leq r$ such that $\gcd(b, r) = 1$. Then there exists a $(b, r)-$fairlet decomposition of X.

Toward fairlets

Lemma 3

Let balance$(X) = b/r$ for some integers $1 \leq b \leq r$ such that $\gcd(b, r) = 1$. Then there exists a $(b, r)-$fairlet decomposition of X.

- This lemma tells us that every fair solution to the clustering problem induces a set of minimal fairlets
- (Proof is very simple! The proof in the paper seems too complex...)

$(t, k)-$fair clustering problems

$(t, k)-$fair center (resp. median) problem

Partition X into $\mathcal{C}$ such that

- $|\mathcal{C}| = k$
- balance$(\mathcal{C}) \geq t$
- $\phi(X, \mathcal{C})$ (resp. $\psi(X, \mathcal{C})$) is minimized.

$(t,k)-$fair clustering problems

$(t,k)-$fair center (resp. median) problem

Partition X into $\mathcal{C}$ such that

- $|\mathcal{C}| = k$
- balance$(\mathcal{C}) \geq t$
- $\phi(X, \mathcal{C})$ (resp. $\psi(X, \mathcal{C})$) is minimized.

- If fairness is not taken into account, the assignment function is implicit through a set $\{c_1, \ldots, c_k\}$ of centers i.e.

$$\alpha(x) = \text{argmin}_{i \in [k]} d(x, c_i)$$

$(t, k)-$fair clustering problems

$(t, k)-$fair center (resp. median) problem

Partition X into $\mathcal{C}$ such that

- $|\mathcal{C}| = k$
- balance$(\mathcal{C}) \geq t$
- $\phi(X, \mathcal{C})$ (resp. $\psi(X, \mathcal{C})$) is minimized.

- If fairness is not taken into account, the assignment function is implicit through a set $\{c_1, \ldots, c_k\}$ of centers i.e.

$$\alpha(x) = \text{argmin}_{i \in [k]} d(x, c_i)$$

- With fairness, an explicit assignment function is required.

Outline

Fairlet decomposition cost

- $\mathcal{Y} = \{Y_1, \ldots, Y_m\}$: a fairlet decomposition of X
- $y_j \in Y_j$ is the *center* of Y_j. (Its choice is arbitrary)
- $\beta : X \to [m]$ is the mapping from a point to the index of the fairlet to which it is mapped.

Definition

For a fairlet decomposition $\mathcal{Y}$, define its costs:

- k-median cost $= \sum_{x \in X} d\left(x, y_{\beta(x)}\right) =: \psi(X, \mathcal{Y})$
- k-center cost $= \max_{x \in X} d\left(x, y_{\beta(x)}\right) =: \phi(X, \mathcal{Y})$

Also, we say that a (b, r)-fairlet decomposition is *optimal* if it has minimum cost among all possible (b, r)-fairlet decompositions.

Reduction to colorblind clustering

- Recall that a (t, k)-fair clustering of X requires that $t \leq \text{balance}(X)$

Reduction to colorblind clustering

- Recall that a (t, k)-fair clustering of X requires that $t \leq \text{balance}(X)$
- To achieve this, we consider the vanilla k-clustering of the centers of each fairlet i.e. k-clustering of $\{y_1, \ldots, y_m\}$

Reduction to colorblind clustering

- Recall that a (t, k)-fair clustering of X requires that $t \leq \text{balance}(X)$
- To achieve this, we consider the vanilla k-clustering of the centers of each fairlet i.e. k-clustering of $\{y_1, \ldots, y_m\}$
- Then we obtain a set of centers $\{c_1, \ldots, c_k\}$ and an assignment function $\alpha_Y : Y \to [k]$.

Reduction to colorblind clustering

- Recall that a (t, k)-fair clustering of X requires that $t \leq \text{balance}(X)$
- To achieve this, we consider the vanilla k-clustering of the centers of each fairlet i.e. k-clustering of $\{y_1, \ldots, y_m\}$
- Then we obtain a set of centers $\{c_1, \ldots, c_k\}$ and an assignment function $\alpha_Y : Y \to [k]$.
- Define $\alpha(x) = \alpha_Y(y_{\beta(x)})$ as the overall assignment function and denote $\mathcal{C}_\alpha$ as the clustering induced by α.

Reduction to colorblind clustering

- Recall that a (t, k)-fair clustering of X requires that $t \leq \text{balance}(X)$
- To achieve this, we consider the vanilla k-clustering of the centers of each fairlet i.e. k-clustering of $\{y_1, \ldots, y_m\}$
- Then we obtain a set of centers $\{c_1, \ldots, c_k\}$ and an assignment function $\alpha_Y : Y \to [k]$.
- Define $\alpha(x) = \alpha_Y(y_{\beta(x)})$ as the overall assignment function and denote $\mathcal{C}_\alpha$ as the clustering induced by α.
- Then we have that balance $C_\alpha = t$

Reduction to colorblind clustering

- Recall that a (t, k)-fair clustering of X requires that $t \leq \text{balance}(X)$
- To achieve this, we consider the vanilla k-clustering of the centers of each fairlet i.e. k-clustering of $\{y_1, \ldots, y_m\}$
- Then we obtain a set of centers $\{c_1, \ldots, c_k\}$ and an assignment function $\alpha_Y : Y \to [k]$.
- Define $\alpha(x) = \alpha_Y(y_{\beta(x)})$ as the overall assignment function and denote $\mathcal{C}_\alpha$ as the clustering induced by α.
- Then we have that balance $C_\alpha = t$
- Also, its cost is bounded, as shown in the next lemma.

Reduction to colorblind clustering

Lemma 6 (corrected)

Denote $\tilde{Y}$ as a multiset where each y_i appears $|Y_i|$ number of times. Then,

$$\psi(X, \mathcal{C}_\alpha) \leq \psi(X, \mathcal{Y}) + \psi(\tilde{Y}, \mathcal{C}_\alpha)$$

$$\phi(X, \mathcal{C}_\alpha) \leq \phi(X, \mathcal{Y}) + \phi(\tilde{Y}, \mathcal{C}_\alpha)$$

Reduction to colorblind clustering

Lemma 6 (corrected)

Denote $\tilde{Y}$ as a multiset where each y_i appears $|Y_i|$ number of times. Then,

$$\psi(X, \mathcal{C}_\alpha) \leq \psi(X, \mathcal{Y}) + \psi(\tilde{Y}, \mathcal{C}_\alpha)$$

$$\phi(X, \mathcal{C}_\alpha) \leq \phi(X, \mathcal{Y}) + \phi(\tilde{Y}, \mathcal{C}_\alpha)$$

This lemma, along with previous reasoning, shows that the fair clustering problem can be reduced to

- Find a good fairlet decomposition (α-approximation)
- Solve the vanilla clustering problem on the centers of the fairlets (β-approximation)

, which is actually a $(\alpha + \beta)$-approximation in total!

Outline

Outline

1. Introduction
2. Previous / Related Works
3. Preliminaries
4. Fairlet decomposition and fair clustering
5. Algorithms
   - (1, k)-fair center problem
   - (1/t', k)-fair center problem
   - (1/t', k)-fair median problem
   - Hardness
6. Experiments
7. Summary / Future Research
8. Summary of Fairness
9. References

Fair k-center: $(1, 1)$-fairlets

- Let us first consider the case when $\text{balance}(X) = 1$.

- Let us first consider the case when $\text{balance}(X) = 1$.
- How can we find a perfectly balanced clustering?

Fair k-center: $(1, 1)$-fairlets

- Let us first consider the case when $\text{balance}(X) = 1$.
- How can we find a perfectly balanced clustering?
- We utilize a good $(1, 1)$-fairlet decomposition!

Fair k-center: $(1, 1)$-fairlets

- Let us first consider the case when $\text{balance}(X) = 1$.
- How can we find a perfectly balanced clustering?
- We utilize a good $(1, 1)$-fairlet decomposition!

Lemma 7

An optimal $(1, 1)$-fairlet decomposition for k-center can be found in polynomial time.

(The approach used in the proof will be used later!)

Proof of Lemma 7

- We shall prove this by relating it to a *graph covering problem*.
- Denote $B(X) = \{b_i\}_i$ and $R(X) = \{r_j\}_j$
- Create a weighted, *complete* bipartite graph $G = (B, R, E)$ with the weight function $w(b_i, r_j) = d(b_i, r_j)$
- Every $(1, 1)$-fairlet decomposition corresponds to some perfect matching in G where each edge represents a fairlet, Y_i.
- Letting $\mathcal{Y} = \{Y_i\}_i$, the k-center cost $\phi(X, \mathcal{Y})$ is exactly the cost of the maximum weight edge in the matching.

Proof of Lemma 7

- Now, our problem is to find a perfect matching that minimizes the weight of the maximum edge.
- Can be done in $O(n^2)$ time.
  (cf. "threshold graph", binary searching)
- For each Y_i, arbitrarily set one of the two nodes of the corresponding edge as the center, y_i.

Fair k-center: $(1, 1)$-fairlets

- Any fair solution induces a set of minimal fairlets. (Lemma 3)

[7] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

Fair k-center: $(1,1)$-fairlets

- Any fair solution induces a set of minimal fairlets. (Lemma 3)
- Thus, the cost of the fairlet decomposition found is at most *twice* the cost of an optimal solution to the clustering.

Lemma 8 (corrected)[16]

Let $\mathcal{Y}$ be the partition found previously, and let ϕ_t^* be the cost of the optimal (t,k)-fair center clustering. Then, $\phi(X,\mathcal{Y}) \leq 2\phi_t^*$.

[7] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

Fair k-center: $(1, 1)$-fairlets

- Any fair solution induces a set of minimal fairlets. (Lemma 3)
- Thus, the cost of the fairlet decomposition found is at most *twice* the cost of an optimal solution to the clustering.

Lemma 8 (corrected)[16]

Let $\mathcal{Y}$ be the partition found previously, and let ϕ_t^* be the cost of the optimal (t, k)-fair center clustering. Then, $\phi(X, \mathcal{Y}) \leq 2\phi_t^*$.

- Let us utilize a result by Gonzalez for k-center problem[7]:

Theorem (Gonzalez, 1985)

There is an algorithm which, given a k-center instance $\mathcal{I}$, produces a 2-approximation solution to $\mathcal{I}$ in running time $O(kn)$

[7] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

Fair k-center: $(1, 1)$-fairlets

Theorem 9 (corrected)

The algorithm that first finds fairlets and then clusters them is a 4-approximation for the $(1, k)$-fair center problem.

Outline

Fair k-center: $(1, t')$-fairlets

- Now let us consider the case when $\text{balance}(X) = t < 1$.

Fair k-center: $(1, t')$-fairlets

- Now let us consider the case when $\text{balance}(X) = t < 1$.
- For simplicity, assume that $t = 1/t'$ for some integer $t' > 1$ (as done in the paper)

Fair k-center: $(1, t')$-fairlets

- Now let us consider the case when $\text{balance}(X) = t < 1$.
- For simplicity, assume that $t = 1/t'$ for some integer $t' > 1$ (as done in the paper)
- As a generalization of previous argument, we shall transform this problem into a minimum cost flow problem (MCFP).

MCFP

Definition

A *flow network* is a directed graph $G = (V, E)$ with a source vertex $s \in V$ and a sink vertex $t \in V$, where each edge $(u, v) \in E$ has capacity $c(u, v) > 0$, flow $f(u, v) \geq 0$ and cost $a(u, v) \in \mathbb{R}$

Minimum Cost Flow Problem (MCFP)

Input: A flow network $(G = (V, E), s, t, c, a)$ (without the flow), d

Constraints:

- Capacity constraints: $f(u, v) \leq c(u, v)$
- Skew symmetry: $f(u, v) = -f(v, u)$
- Flow conservation: $\forall u \neq s, t \ \sum_{w \in V} f(u, w) = 0$
- Required flow from s to t: $\sum_{w \in V} f(s, w) = \sum_{w \in V} f(w, t) = d$

Output: Flow $f(u, v)$ such that $\sum_{(u,v) \in E} a(u, v) f(u, v)$ is minimized

Fair k-center: $(1, t')$-fairlets

- Let us construct an instance of MCFP, with a parameter $\tau > 0$.

Fair k-center: $(1, t')$-fairlets

- Let us construct an instance of MCFP, with a parameter $\tau > 0$.
- First, let us construct a directed graph $H_\tau = (V, E)$

Fair k-center: $(1, t')$-fairlets

- Let us construct an instance of MCFP, with a parameter $\tau > 0$.
- First, let us construct a directed graph $H_\tau = (V, E)$
- Vertex set:

$$V = \{\beta, \rho\} \cup B(X) \cup R(X) \cup \left\{ b_i^j | b_i \in B(X) \right\}_{j \in [t']} \cup \left\{ r_i^j | r_i \in R(X) \right\}_{j \in [t']}$$

Fair k-center: $(1, t')$-fairlets

- Let us construct an instance of MCFP, with a parameter $\tau > 0$.
- First, let us construct a directed graph $H_\tau = (V, E)$
- Vertex set:

$$V = \{\beta, \rho\} \cup B(X) \cup R(X) \cup \left\{ b_i^j | b_i \in B(X) \right\}_{j \in [t']} \cup \left\{ r_i^j | r_i \in R(X) \right\}_{j \in [t']}$$

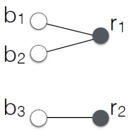
- Edge set:
  - (β, ρ) with cost 0 and capacity $\min(|B(X)|, |R(X)|)$
  - (β, b_i) and (r_i, ρ) for each $b_i \in B(X), r_i \in R(X)$, each with cost 0 and capacity $t' - 1$
  - (b_i, b_i^j) and (r_i, r_i^j) for each $b_i \in B(X), r_i \in R(X), j \in [t']$, each with cost 0 and capacity 1
  - (b_i^k, r_j^l) for each $b_i \in B(X), r_i \in R(X), 1 \leq k, l \leq t'$, each with cost 1 if $d(b_i, r_j) \leq \tau$ and ∞ otherwise.

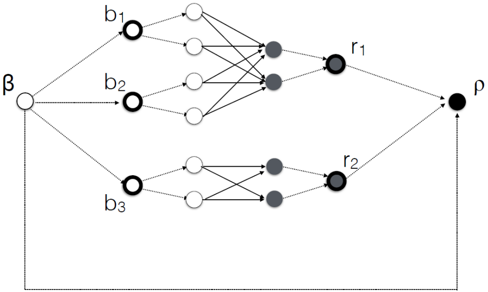
Fair k-center: $(1, t')$-fairlets

- To finish the description, we need specify the supply and demand at every node:
  - Every node in $B(X)$ has a supply of 1
  - Every node in $R(X)$ has a demand of 1
  - β has a supply of $|R(X)|$
  - ρ has a demand of $|B(X)|$
  - Every other node has zero supply and demand

Fair k-center: $(1, t')$-fairlets

- To finish the description, we need specify the supply and demand at every node:
  - Every node in $B(X)$ has a supply of 1
  - Every node in $R(X)$ has a demand of 1
  - β has a supply of $|R(X)|$
  - ρ has a demand of $|B(X)|$
  - Every other node has zero supply and demand





Fair k-center: $(1, t')$-fairlets

Lemma 10

Let $\mathcal{Y}$ be a $(1, t')$-fairlet decomposition of cost C for the $(1/t', k)$-fair center problem. Then it is possible to construct a feasible solution of cost $2C$ to the (constructed) MCF instance.

Above lemma tells us that a $(1, t')$-fairlet decomposition can be used to construct a feasible solution for the MCF instance of twice the cost.

Fair k-center: $(1, t')$-fairlets

Lemma 11

Let $\mathcal{Y}$ be an optimal solution of cost C to the (constructed) MCF instance. Then it is possible to construct a $(1, t')$-fairlet decomposition for the $(1/t', k)$-fair center problem of cost at most C.

Above lemma tells us that an optimal solution for the MCF instance can be used to obtain a $(1, t')$-fairlet decomposition of bounded cost.

Fair k-center: $(1, t')$-fairlets

Combining the previous two lemmas yield:

Lemma 12

By reducing the $(1, t')$-fairlet decomposition problem to an MCFP, it is possible to compute a 2-approximation for the optimal $(1, t')$-fairlet decomposition for the $(1/t', k)$-fair center problem.

Combining above with the result by Gonzalez gives... (next slide)

Fair k-center

Theorem 13

For any integer $t' \in \mathbb{N}$, the algorithm that first finds fairlets and then clusters them is a 4-approximation for the $(1/t', k)$-fair center problem.

Outline

Fair k-median

- Slight modification of previous argument gives us the (approx.) solution for (t, k)-fair median problem (with $t = 1/t'$)

[8] Shi Li and Ola Svensson. "Approximating k-median via pseudo-approximation". In: *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013.* 2013, pp. 901–910.

Fair k-median

- Slight modification of previous argument gives us the (approx.) solution for (t, k)-fair median problem (with $t = 1/t'$)
- For the perfectly balanced case, our goal is to look for a perfect matching of minimum total cost on the bichromatic graph.

[8] Shi Li and Ola Svensson. "Approximating k-median via pseudo-approximation". In: *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013.* 2013, pp. 901–910.

- Slight modification of previous argument gives us the (approx.) solution for (t, k)-fair median problem (with $t = 1/t'$)
- For the perfectly balanced case, our goal is to look for a perfect matching of minimum total cost on the bichromatic graph.
- To find $(1, t')$-fairlet decomposition for $t' > 1$, create an instance of MCF, *with (some of the) weights as the distances.*

[8] Shi Li and Ola Svensson. "Approximating k-median via pseudo-approximation". In: *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013.* 2013, pp. 901–910.

Fair k-median

- Slight modification of previous argument gives us the (approx.) solution for (t, k)-fair median problem (with $t = 1/t'$)
- For the perfectly balanced case, our goal is to look for a perfect matching of minimum total cost on the bichromatic graph.
- To find $(1, t')$-fairlet decomposition for $t' > 1$, create an instance of MCF, *with (some of the) weights as the distances*.
- Let us utilize a result by Li & Svensson for k-median problem[8]:

Theorem (Li & Svensson, 2013)

There is an algorithm which, given a k-median instance $\mathcal{I}$ and $\varepsilon > 0$, produces a $(1 + \sqrt{3} + \varepsilon)$-approximation solution to $\mathcal{I}$ in running time $O\left(n^{O(1/\varepsilon^2)}\right)$

[8] Shi Li and Ola Svensson. "Approximating k-median via pseudo-approximation". In: *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013.* 2013, pp. 901–910.

Fair k-median

Theorem 15

For any integer $t' \in \mathbb{N}$, the algorithm that first finds fairlets and then clusters them is a $(t' + 1 + \sqrt{3} + \varepsilon)$-approximation for the $(1/t', k)$-fair median problem.

Outline

Hardness

- We now have a theoretical framework and an actual algorithm for solving fair clustering problems.

Hardness

- We now have a theoretical framework and an actual algorithm for solving fair clustering problems.
- But by taking fairness into account, we have introduced some extra complexity to the classical clustering problems.

Hardness

- We now have a theoretical framework and an actual algorithm for solving fair clustering problems.
- But by taking fairness into account, we have introduced some extra complexity to the classical clustering problems.
- How bad can it be, right?

Hardness

- We now have a theoretical framework and an actual algorithm for solving fair clustering problems.
- But by taking fairness into account, we have introduced some extra complexity to the classical clustering problems.
- How bad can it be, right?
- Well, as the next theorem shows, ensuring fairness actually introduces a computational bottleneck! (and a very narrow one, indeed.)

Hardness

- We now have a theoretical framework and an actual algorithm for solving fair clustering problems.
- But by taking fairness into account, we have introduced some extra complexity to the classical clustering problems.
- How bad can it be, right?
- Well, as the next theorem shows, ensuring fairness actually introduces a computational bottleneck! (and a very narrow one, indeed.)

Theorem 16

For each fixed $t' \geq 3$,

- Finding an optimal $(1, t')$-fairlet decomposition is NP-hard.
- Finding the minimum cost $(1/t', k)$-fair median clustering is NP-hard

Outline

Experiments

The goal of the experiments is two-fold:

Experiments

The goal of the experiments is two-fold:

- Show that the traditional algorithms for k-center and k-median tend to produce unfair clusters

Experiments

The goal of the experiments is two-fold:

- Show that the traditional algorithms for k-center and k-median tend to produce unfair clusters
- Show that the proposed algorithm outputs clusters that respect the fairness guarantees

Experiment Design

- Datasets used: **Diabetes, Bank, Sensus**[9]
  (Protected attributes: **gender, married or not, gender**, respectively)

[9] Mosche Lichman and Kevin Bache. *UCI Machine Learning Repository*. 2013. URL: http://archive.ics.uci.edu/ml.

[10] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

[11] Vijay Arya et al. "Local Search Heuristics for k-Median and Facility Location Problems". In: *SIAM J. Comput.* 33.3 (2004), pp. 544–562.

- Datasets used: **Diabetes, Bank, Sensus**[9]
  (Protected attributes: **gender, married or not, gender**, respectively)
- Flow-based fairlet decomposition algorithm (as proposed) was implemented.

[9] Mosche Lichman and Kevin Bache. *UCI Machine Learning Repository*. 2013. URL: http://archive.ics.uci.edu/ml.

[10] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

[11] Vijay Arya et al. "Local Search Heuristics for k-Median and Facility Location Problems". In: *SIAM J. Comput.* 33.3 (2004), pp. 544–562.

Experiment Design

- Datasets used: **Diabetes, Bank, Sensus**[9]
  (Protected attributes: **gender, married or not, gender**, respectively)
- Flow-based fairlet decomposition algorithm (as proposed) was implemented.
- For the vanilla k-center clustering algorithm, the *greedy furthest point algorithm*[10] was used.
  (known to obtain 2-approximation)

[9] Mosche Lichman and Kevin Bache. *UCI Machine Learning Repository*. 2013. URL: http://archive.ics.uci.edu/ml.

[10] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

[11] Vijay Arya et al. "Local Search Heuristics for k-Median and Facility Location Problems". In: *SIAM J. Comput.* 33.3 (2004), pp. 544–562.

Experiment Design

- Datasets used: **Diabetes, Bank, Sensus**[9]
  (Protected attributes: **gender, married or not, gender**, respectively)
- Flow-based fairlet decomposition algorithm (as proposed) was implemented.
- For the vanilla k-center clustering algorithm, the *greedy furthest point algorithm*[10] was used.
  (known to obtain 2-approximation)
- For the vanilla k-median clustering algorithm, *single swap algorithm*[11] was used.
  (known to obtain 5-approximation in the worst case, but performs well in practice. Refer to Kanungo *et al.*, 2002)

---

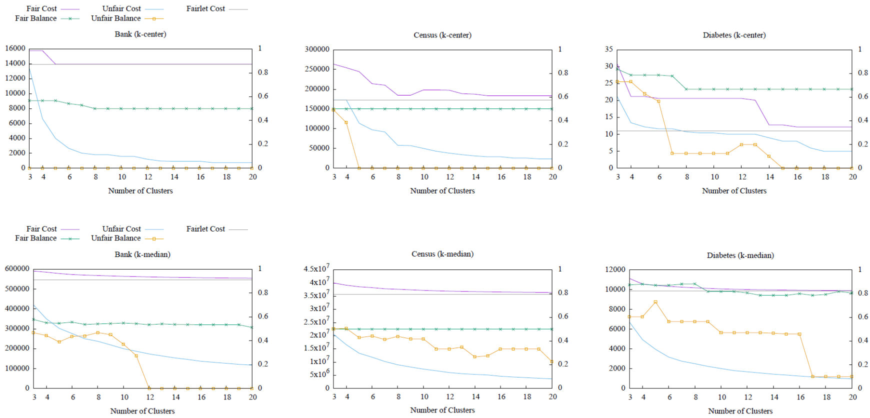
[9] Mosche Lichman and Kevin Bache. *UCI Machine Learning Repository*. 2013. URL: http://archive.ics.uci.edu/ml.

[10] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

[11] Vijay Arya et al. "Local Search Heuristics for k-Median and Facility Location Problems". In: *SIAM J. Comput.* 33.3 (2004), pp. 544–562.

Results

In all cases, the experiment was done with $t' = 2$ i.e. aiming for balance of at least 0.5 in each cluster.



Outline

Summary

In summary,

Summary

In summary,

- Reduction of fair clustering to classical clustering via fairlets

Summary

In summary,

- Reduction of fair clustering to classical clustering via fairlets
- Efficient approximation algorithms for finding fairlet decompositions

Summary

In summary,

- Reduction of fair clustering to classical clustering via fairlets
- Efficient approximation algorithms for finding fairlet decompositions
- Showed that fairness can introduce a computational bottleneck

Future research

- Improve the approximation ratio of the decomposition algorithms

Future research

- Improve the approximation ratio of the decomposition algorithms
- Give stronger hardness results

Future research

- Improve the approximation ratio of the decomposition algorithms
- Give stronger hardness results
- Extend to the case where the protected class is not binary, but can take on multiple values
  (Already done! *Scalable Fair Clustering (ICML 2019)*)

Outline

Summary of Fairness

Here is a summary of "major" concepts in fairness[12][13]: (next page)

[12] Ninareh Mehrabi et al. "A Survey on Bias and Fairness in Machine Learning". In: *arXiv e-prints* (Aug. 2019). arXiv: 1908.09635 [cs.LG].

[13] Ziyuan Zhong. *A Tutorial on Fairness in Machine Learning*. July 2019. URL: https://towardsdatascience.com/a-tutorial-on-fairness-in-machine-learning-3ff8ba1040cb.

Summary of Fairness

Fairness Metrics	Key Features	Pros	Cons
Equalized Odds**[10]**	$\forall y \in \{0,1\}$ $P[E=1 \mid A=0, Y=y] = P[E=1 \mid A=0, Y=y]$	Optimality compatibility: E=Y is allowed. Penalize laziness: it provides incentive to reduce errors uniformly in all groups.	It may not help closing the gap between two groups.
Equal Opportunity**[10]**	$P[E=1 \mid A=0, Y=1] = P[E=1 \mid A=1, Y=1]$	Optimality compatibility: E=Y is allowed. Penalize laziness: it provides incentive to reduce errors uniformly in all groups.	It may not help closing the gap between two groups.
Demographic (Statistical) Parity**[4]**	$P[E=1 \mid A=0] = P[E=1 \mid A=1]$	Legal support (80%-rule, **adverse impact**)	This definition ignores any correlation between Y and A
Conditional Statistical Parity**[4]**	$P[E=1 \mid L=1, A=0] = P[E=1 \mid L=1, A=1]$	-	-
Fairness Through Awareness**[4]**	Similar predictions to similar individuals (cf. Lipschitz condition)	It emphasizes more about the individuals and imposes restriction on the treatment for each pair of individuals.	It is hard to determine the appropriate metric.
Fairness Through Unawareness[8]	No protected attributes is explicitly used in the decision-making process	Intuitive, easy to use and legal support(**disparate treatment**).	There can be many highly correlated features.
Treatment Equality**[18]**	Ratio of false negative and false positives is the same for both protected group categories	Optimality compatibility: E=Y is allowed. Equal chance of success(Y=1) given acceptance(E=1).	It may not help closing the gap between two groups.
Counterfactual Fairness**[12]**	$P[E_{A\leftarrow a}(U) = y \mid X=x, A=a] = P[E_{A\leftarrow a'}(U) = y \mid X=x, A=a]$	It provides a way of explaining the impact of bias via a causal graph.	It is hard to agree on the causal graph and to decide which features to use.
Fairness in Relational Domains**[5]**	Also takes into account the social, organizational, and other connections between individuals	-	-

Outline

References I

[1] Vijay Arya et al. "Local Search Heuristics for k-Median and Facility Location Problems". In: *SIAM J. Comput.* 33.3 (2004), pp. 544–562.

[2] Dan Biddle. *Adverse Impact and Test Validation: A Practitioner's Guide to Valid and Defensible Employment Testing*. 2nd ed. Routledge, 2006.

[3] Flavio Chierichetti et al. "Fair Clustering Through Fairlets". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*. 2017, pp. 5029–5037.

[4] Cynthia Dwork et al. "Fairness through awareness". In: *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*. 2012, pp. 214–226.

[5] Golnoosh Farnadi, Behrouz Babaki, and Lise Getoor. "Fairness in Relational Domains". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2018, New Orleans, LA, USA, February 02-03, 2018*. 2018, pp. 108–114.

[6] Michael Feldman et al. "Certifying and Removing Disparate Impact". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*. 2015, pp. 259–268.

[7] Teofilo F. Gonzalez. "Clustering to Minimize the Maximum Intercluster Distance". In: *Theoretical Computer Science* 38 (1985), pp. 293–306.

[8] Nina Grgić-Hlača et al. "The Case for Process Fairness in Learning: Feature Selection for Fair Decision Making". In: *Symposium on Machine Learning and the Law at the 29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain*. 2016.

[9] *Griggs v. Duke Power Co., 401 U.S. 424 (Supreme Court of the United States 1971).*

[10] Moritz Hardt, Eric Price, and Nati Srebro. "Equality of Opportunity in Supervised Learning". In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*. 2016, pp. 3315–3323.

References II

[11] Toshihiro Kamishima et al. "Fairness-Aware Classifier with Prejudice Remover Regularizer". In: *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II*. 2012, pp. 35–50.

[12] Matt J. Kusner et al. "Counterfactual Fairness". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*. 2017, pp. 4066–4076.

[13] Shi Li and Ola Svensson. "Approximating k-median via pseudo-approximation". In: *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*. 2013, pp. 901–910.

[14] Mosche Lichman and Kevin Bache. *UCI Machine Learning Repository*. 2013. URL: http://archive.ics.uci.edu/ml.

[15] Ninareh Mehrabi et al. "A Survey on Bias and Fairness in Machine Learning". In: *arXiv e-prints* (Aug. 2019). arXiv: 1908.09635 [cs.LG].

[16] Clemens Rösner. "Constrained Clustering Problems and Parity Games". Dissertation zur Erlangung des Doktorgrades (Dr. rer. nat.) der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn. Universität Bonn, 2019.

[17] *Uniform Guidelines on Employee Selection Procedures*.

[18] Muhammad Bilal Zafar et al. "Fairness Constraints: Mechanisms for Fair Classification". In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*. 2017, pp. 962–970.

[19] Richard S. Zemel et al. "Learning Fair Representations". In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. 2013, pp. 325–333.

[20] Ziyuan Zhong. *A Tutorial on Fairness in Machine Learning*. July 2019. URL: https://towardsdatascience.com/a-tutorial-on-fairness-in-machine-learning-3ff8ba1040cb.

Thank you for your attention! Any questions?