ALGORITHMIC FAIRNESS IN CLUSTERING: A STUDY IN REPLICATION

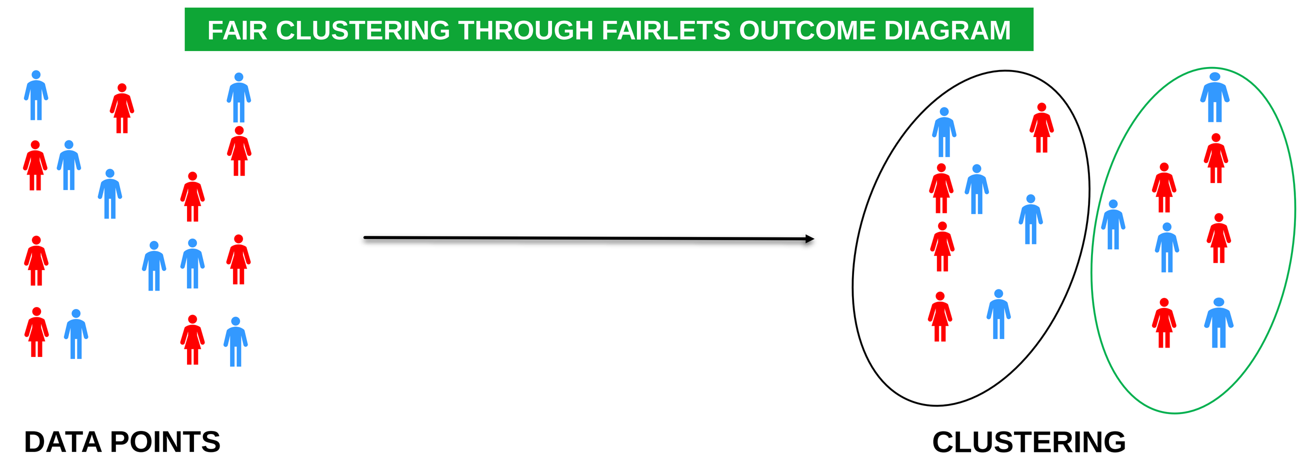
Muno Siyakurima, Sophie Boileau, Avery Hall, Victor Huang, Jeremiah Mensah, Armira Nance, Brie Sloves

Advised by Layla Oesper, Carleton College Computer Science Comps 2024

INTRODUCTION

- Clustering algorithms take in a collection of data (with no labels) and are able to identify groupings of items in the data that share similar features. Numerous clustering algorithms exist. But one popular such algorithm is the K-means clustering algorithm (along with related algorithms K-medians and K-medoids).
- This algorithm aims to minimize the total variance amongst items clustering together, for a provided number of clusters. This approach has been used in a number of different domains including document clustering, identifying regions of cities with higher crime rates, or identifying cancer patients with different molecular profiles. However, in recent years there has been in increased focus on whether such clustering can be considered "fair" when considering certain subgroups in the data (e.g., demographic groups like race and gender)
- **For our project we used a definition of fairness from the paper "Fair Clustering through Fairlets" which states that Fair Clusters are those that maintain the same attribute ratio as the original dataset e.g if gender is the target attribute the diagram below shows a good clustering:**

FAIR CLUSTERING THROUGH FAIRLETS OUTCOME DIAGRAM



OBJECTIVES

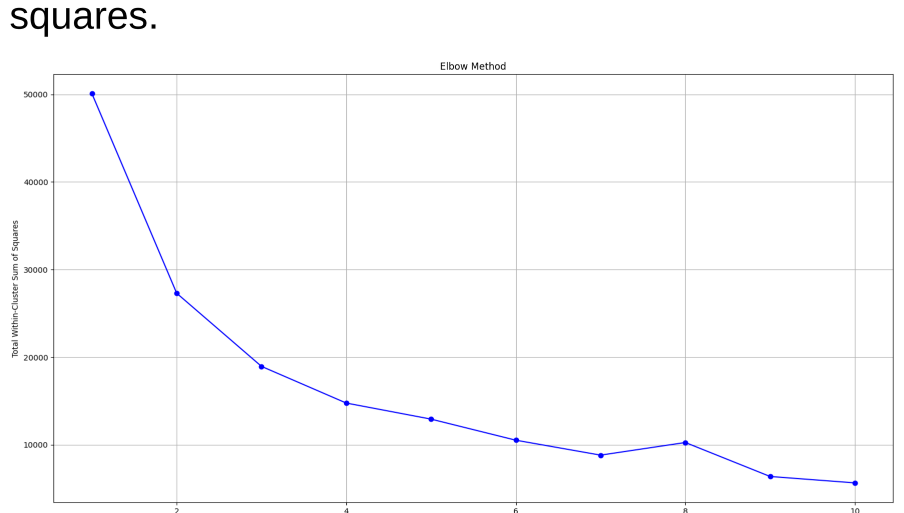
- The objectives of this study involve implementing various clustering algorithms—K-means, K-means++, Basic Fairlets, and Minimum Cost Flow (MCF) Fairlets—on our dataset. We aim to analyze how these algorithms cluster the data and subsequently focus on balancing these clusters, considering specific protected classes such as race and gender.
- Use these algorithms on our own dataset which is the **"High School Longitudinal Study of 2009 (HSLS:09)"** which has data about student performance and demographic information.

METHODS

- Primarily we focused on implementing the basic K-means and the K-means++ algorithms. On top of that we also wanted to replicate the Fairlets code and implement it on our dataset to see if we do get fair results.
- My main task was implementing the basic K-means algorithm aka Lloyd's Algorithm and the Elbow Method.

ELBOW METHOD

- The Elbow Method helps determine the optimal number of clusters by identifying the point where the rate of decrease in variance flattens, resembling an 'elbow' on a graph of cluster numbers versus within-cluster sum of squares.



RESULTS

- As shown in the Methods Section, my Elbow Method implementation shows that the best value for k for our dataset was 5 so all clustering in the project was done with the number of clusters equal to 5.
- For conducting our clustering, we used 4 variables from our dataset:
  - **Socioeconomic Status**
  - **Annual Income per Household Member**
  - **Highest Parent Education Level**
  - **Weekly Hours of Extracurricular Activity**
- For our balancing attribute we chose to look at the attribute **Gender**.
- Additionally, since we were using 4 variables, we had to make use of Principle Component Analysis which allows us to visualize variables of more than 3 dimensions in 2 or 3 dimensions.

BASIC K-MEANS ALGORITHM/ LLOYD'S ALGORITHM

- The Basic K-Means minimizes the sum of squared distances between data points and their respective cluster centroids, aiming to find the centroids that minimize intra-cluster variance and maximize inter-cluster variance.
- The algorithm's performance can be sensitive to the initial placement of centroids, potentially converging to local optima based on these initial points. Multiple initializations can help mitigate this issue.
- As shown below we ran our algorithm 10 times and 60% of the time we got the results shown in Fig. 1 and 20% of the time we got Fig. 2 results

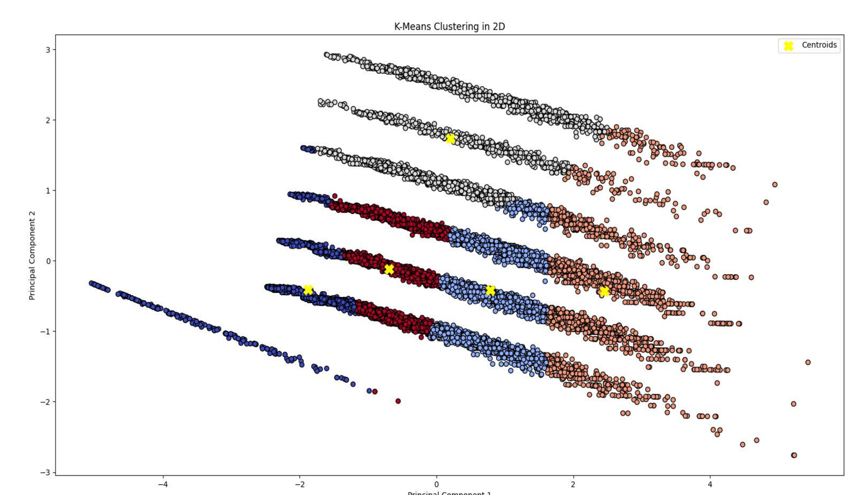


Fig. 1 Example of K-Means Clustering of our data

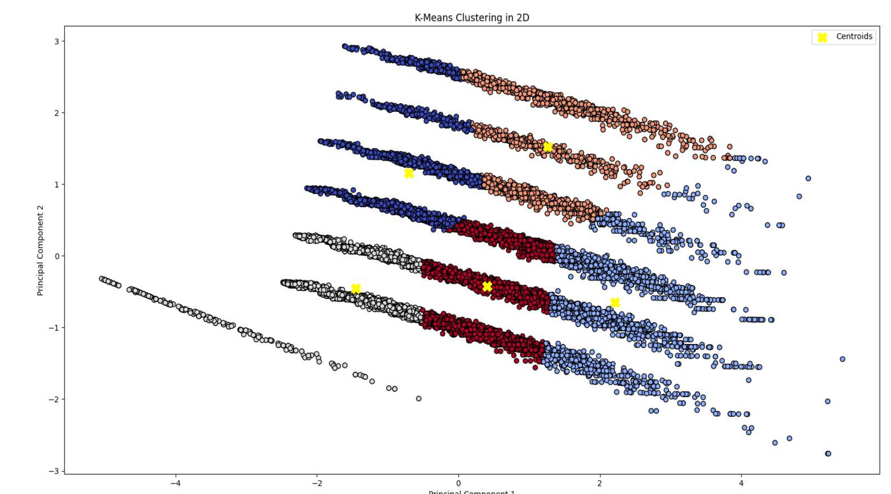


Fig. 2 Example of K-Means Clustering of our data

K-MEANS++ ALGORITHM

- K-means++ is an improvement over the classic K-means algorithm in terms of initialization. It selects initial cluster centroids in a smarter way, reducing the chance of poor convergence by picking centroids that are more spread out in the feature space.
- K-means++ aims to address the randomness in centroid initialization in K-means by employing a probabilistic method. It chooses the initial centroids by considering the distances of points, ensuring a more even spread of initial cluster centers.
- This initialization leads to a faster convergence rate and often results in better overall clustering. Results of the K-Means++ clustering on our data is shown in Fig. 3.

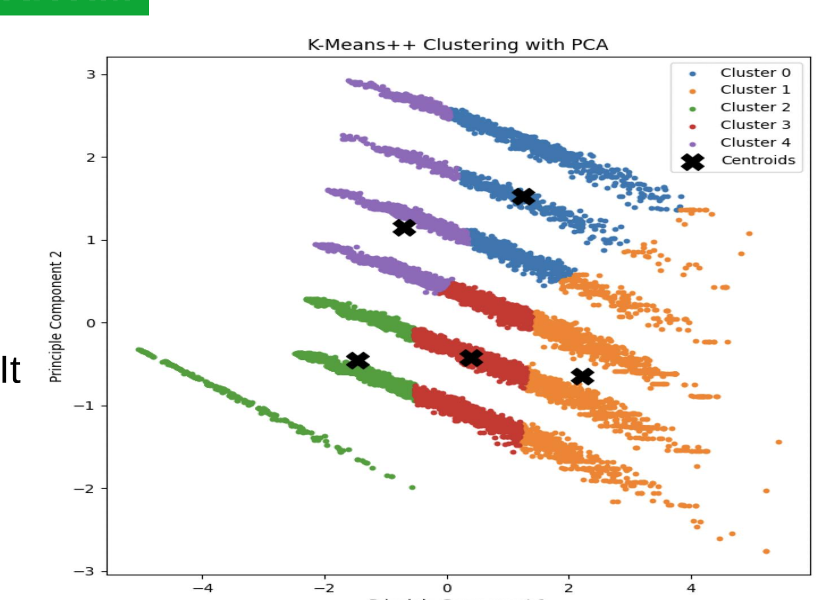


Fig. 3, Plot of K-Means++ Clustering on our data
Credit: Jeremiah Mensah

BALANCE COMPARISONS WITH THE FAIR CLUSTERING THROUGH FAIRLETS ALGORITHM

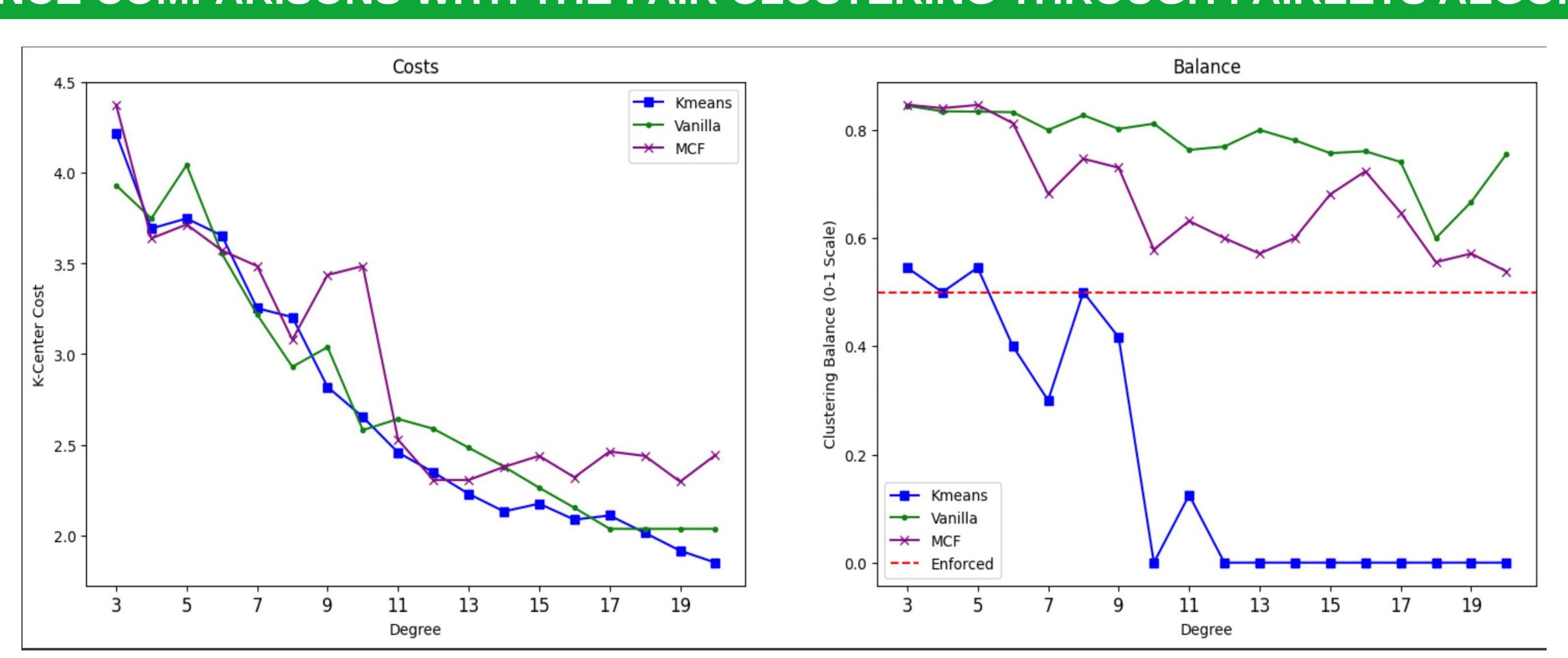


Fig. 4, Plots of K-Center Cost and Clustering Balance of the Different Algorithms
Credit: Victor Huang

CONCLUSIONS

- Looking at the clusters produced by the Basic K-Means and the K-Means++ we can see that there is a slight difference between the two and Fig. 2 of the Basic K-Means is even the same as the K-Means++ output. This is probably because of the initial random initialization step done in the two algorithms and depending on the initial centroids, both algorithms could produce the same results.
- Therefore, when comparing these 2 algorithms with the Basic Fairlets and MCF Fairlets Algorithm in Fig. 4 we utilized the Basic K-Means only.

FAIRNESS COMPARISONS

- From the results shown in Fig. 4 we can see that the two implementations of the Fairlets Algorithm produced balanced clusterings with our balance variable being Sex.
- Additionally, the Basic K-Means algorithm did not perform well in balancing our clusters according to our fairness definition.
- We can also see that the two Fairlets algorithm implemetations also had little effect on the k-cost.
- These results led us to conclude that the Fairlets Algorithm does actually balance clusters very well depending on our balance variable unlike the Basic K-Means and K-Means++ algorithms.

FUTURE RESEARCH

- **Intersectionality:** In our project we used balance attributes that had our data points fall into one of the available categories/protected attributes. However, we know that attributes like Sex which we used in this project is not binary and we have attributes like race were people can fall into more than one category.
  Therefore, it would be interesting to do more research on clustering data with intersectionality traits.
- **Social Fairness Algorithm:** Since for our project our balancing attribute was Gender, it would be interesting to see the results of an algorithm that is particularly meant to balance clusters on social variables like these and see how good it performs compared to the Fairlets Algorithm.

REFERENCES

GitHub Repository – https://github.com/riceboi732/CS-Comps

"High School Longitudinal Study of 2009 (HSLS:09) - Overview." National Center for Education Statistics, nces.ed.gov/surveys/hsls09/. Accessed 5 Nov. 2023.

Chhabra, Anshuman et al. "An Overview of Fairness in Clustering." IEEE, vol. 9, 2021, https://ieeexplore.ieee.org/document/9541160

Chierichetti, Flavio et al. "Fair Clustering Through Fairlets." 31st Conference on Neural Information Processing Systems, 2017,. https://proceedings.neurips.cc/paper_files/paper/2017/file/978fce5bcc4eccc88ad48ce3914124a2-Paper.pdf

Chouldechova, Alexandra and Roth, Aaron. "A Snapshot of the Frontiers of Fairness in Machine Learning." Communications of the ACM, vol. 63, no. 5, 2020, https://cacm.acm.org/magazines/2020/5/244336-a-snapshot-of-the-frontiers-of-fairness-in-machine-learning/fulltext?mobile=false.

ACKNOWLEDGEMENTS

- Layla Oesper for guidance through the project.
- Special thanks to my dedicated groupmates for their unwavering support and collaborative efforts throughout this project.