CONCLUSIONS

From the results shown in Fig. 4 we can see that the two implementations of the Basic K-Means++ algorithms produced balanced clusterings with our balance attribute. 

• Additionally, the Basic K-Means algorithm did not perform well in balancing our clusters according to our fairness definition.

Therefore, we can see that the two K-Means implementations 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.

FAIRNESS COMPARISONS

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 an 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:

<table>
<thead>
<tr>
<th>DATA POINTS</th>
<th>CLUSTERING</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com/kmeans.png" alt="Example of K-Means Clustering of our data" /></td>
<td><img src="https://example.com/kmeans++.png" alt="Example of K-Means++ Clustering of our data" /></td>
</tr>
</tbody>
</table>

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 dataset which is the “High School Longitudinal Study of 2009 (HLSL: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.

- 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 clustering 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.

<table>
<thead>
<tr>
<th>BASIC K-MEANS ALGORITHM/ LLOYD’S ALGORITHM</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com/kmeans.png" alt="Example of K-Means Clustering of our data" /></td>
</tr>
</tbody>
</table>

- The Basic K-Means Algorithm 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.

<table>
<thead>
<tr>
<th>K-MEANS++ ALGORITHM</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com/kmeans++.png" alt="Example of K-Means++ Clustering with PCA" /></td>
</tr>
</tbody>
</table>

- 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.

<table>
<thead>
<tr>
<th>BALANCE COMPARISONS WITH THE FAIR CLUSTERING THROUGH FAIRLETS ALGORITHM</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com/kmeans++.png" alt="Plot of K-Means++ Clustering on our data" /></td>
</tr>
</tbody>
</table>

- 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.

<table>
<thead>
<tr>
<th>CENTER COST AND CLUSTERING BALANCE OF THE DIFFERENT ALGORITHMS</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com/center_cost.png" alt="Plot of Center Cost and Clustering Balance of the Different Algorithms" /></td>
</tr>
</tbody>
</table>

- From the results shown in Fig. 4 we can see that the two implementations of the Fairlets Algorithm produced balanced clusterings with our balance attribute. 

- Additionally, the Basic K-Means algorithm did not perform well in balancing our clusters according to our fairness definition.

Therefore, we can see that the two K-Means implementations 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

GitHub Repository: https://github.com/vicb07302/CS-Comps


REFERENCES

ACKNOWLEDGMENTS

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

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

Advised by Layla Oesper, Carleton College Computer Science Comps 2024