

Universiteit Leiden

# Master Computer Science

Fairness across Community Sizes in Network Anonymization


| | |
|---|---|
| Name: | Fabian Jennrich |
| Student ID: | s4055225 |
| Date: | 09/06/2026 |
| Specialisation: | Data Science |
| 1st supervisor: | Dr. Akrati Saxena |
| 2nd supervisor: | Prof. Dr. Frank Takes |


Master's Thesis in Computer Science


Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

**Abstract**

Network anonymization aims to prevent malicious actors from discovering the personal or identifying information of individuals in a network. When publishing a network there is often a lot of personal identifying information about users attached, which we want to remain hidden from malicious actors. This is done via network anonymization. We focus in this thesis on network anonymization methods which rely on making changes to the structure of the published network. Since anonymization is critical for ensuring the safety of users when a network is published, it is important that anonymization procedures do not introduce new biases, or strengthen existing ones. Fairness in social network analysis (SNA) is the idea that all nodes and all types of nodes should be treated equally. We define fairness for network anonymization, and propose methods to measure it. We investigate the fairness with regard to attribute induced minority and majority communities of 19 network anonymization methods. We take a black box approach to the algorithms, and compute fairness based on the anonymous graphs they produce. We investigate fairness in utility and anonymity on real world networks, on synthetic networks with differently sized minority communities, and on synthetic networks with more than two communities. We find that for all the anonymization algorithms we consider, there is unfairness in privacy with favor towards smaller communities. We find that results in utility are heavily dependent on the anonymization method and graph being anonymized, and we do not see a consistent trend of unfairness in any algorithm or graph type. We found that all of the algorithms that we investigated had better privacy results in minority communities, and most had better utility results in majority communities.

# Contents


**1 Introduction** **3**
- 1.1 Contributions . . . 4
- 1.2 Thesis Overview . . . 4

**2 Related Works** **5**
- 2.1 Fairness in SNA . . . 5
- 2.2 Network anonymization . . . 6
- 2.3 Fairness in Network Anonymization . . . 7

**3 Preliminaries** **8**
- 3.1 Graph Properties . . . 8
- 3.2 Graph Metrics . . . 9

**4 Methodology** **10**
- 4.1 Anonymization Algorithms . . . 10
  - 4.1.1 ANO-NET . . . 10
  - 4.1.2 A New Method for Adding Noise Nodes . . . 10
  - 4.1.3 k-Degree Sequence Anonymization . . . 10
  - 4.1.4 Pseudo Node Anonymization . . . 11
  - 4.1.5 k-Degree Anonymity with Vertex and Edge Modification . . . 11
  - 4.1.6 Two Step Anonymization . . . 11
  - 4.1.7 Differential Privacy with Field Theory . . . 11
  - 4.1.8 Edge Insertion Differential Privacy . . . 12
  - 4.1.9 Privacy Preservation Based on Clustering and Noise . . . 12
  - 4.1.10 Subgraph Differential Privacy . . . 12
  - 4.1.11 Optimization via Link Perturbation using Link Importance . . . 12
  - 4.1.12 Edge Entropy . . . 12
  - 4.1.13 Link Perturbation . . . 13
  - 4.1.14 Link Mirage . . . 13
  - 4.1.15 Random Walk . . . 13
  - 4.1.16 Upper Approximation based Anonymization . . . 13
  - 4.1.17 Friend in the Middle . . . 13
- 4.2 Datasets . . . 14
  - 4.2.1 Facebook . . . 14
  - 4.2.2 Twitter . . . 14
  - 4.2.3 Antelope Valley . . . 14
  - 4.2.4 HBA . . . 14
  - 4.2.5 Community Designation . . . 15

4.2.6 Ground Truth Graph Structures . . . . . 16
4.3 Utility Assessment . . . . . 19
4.4 Privacy Assessment . . . . . 20
4.5 Fairness Assessment . . . . . 20
4.6 Experimental Setup . . . . . 21
4.6.1 Implementation of the Algorithms . . . . . 21
4.6.2 Details on Running Testing . . . . . 23

**5 Results** **24**
5.1 Fairness on Real World Social Networks . . . . . 24
5.1.1 Privacy Fairness on Real World Social Networks . . . . . 24
5.1.2 Utility Fairness on Real World Social Networks . . . . . 26
5.2 Fairness in Different Sized Minorities . . . . . 31
5.2.1 Privacy Fairness over Different Community Sizes . . . . . 31
5.2.2 Utility Fairness over Different Community Sizes . . . . . 32
5.3 Fairness with Multiple Minority Communities . . . . . 36
5.3.1 Privacy Fairness with Multiple Minority Communities . . . . . 36
5.3.2 Utility Fairness with Multiple Minority Communities . . . . . 37
5.4 Results Summary . . . . . 39

**6 Conclusion** **43**
6.1 Future Research . . . . . 43

# Chapter 1

# Introduction

Social networks are widely used, and analyzing them provides valuable insights into both individual behavior and patterns of interaction among people. Social network analysis (SNA) uses structure and other network information to understand these interactions and allows for the design of more effective interventions in both online and real life social networks, such as campaigns to distribute information, or stopping the spread of a disease. To be able to do analysis on a network, the network has to be published. Since in many social networks, users publish information about themselves, personal information may then be revealed to malicious actors if the network is made public.

The goal of network anonymization is to be able to preserve the safety of sensitive information for users, while still allowing for useful analysis of network data. Sensitive information in a network may be the participation of a user, for instance whether someone is a patient at a specific medical clinic, the link between users, such as matches on a dating site, or attribute information of a user, such as name, age, or location posted on social media. In general when networks are published for analysis, identification information such as names are removed, or replaced with new identifiers. This is called Naive anonymization and allows for the most accurate analysis as all of the structural information remains intact. However, this does not guarantee privacy, as de-anonymization procedures can use node attributes, and the structure of a network to re-associate nodes with their original identifiers [20]. To provide privacy for the individuals who make up a network, the potential for structural and attribute re-identification must be taken into account when networks are published. Anonymizing graphs inherently involves a tradeoff between the privacy of users, and the utility of the graph, since utility for analysis depends on an anonymous graph being representative of the original data, and privacy depends on it not being too directly representative. There are a number of ways to address this issue [30], such as by modifying the graph [6, 27, 39], generalizing nodes into groups [17, 53], or by publishing only computed graph statistics.

Social networks mirror real world interactions and patterns of behavior, so real world communities and groups are mirrored in online social networks. Because of the fact that people interact more with people like them, called homophily in networks, these communities and groups are often split along categories such as gender, race, or language, and some of these groups will be larger than others. Network communities can be determined by node attribute values, or by identifying closely linked groups of

nodes. It is important for analysis and interventions to produce equal quality results for all users, thus the investigation into fairness. When SNA techniques optimize for the network as a whole, smaller community structures can be neglected, leading to minority groups receiving less benefit. We call the largest group in a network the majority community, and refer to all smaller communities as minorities. A number of methods have been proposed to measure fairness in different areas of network science [49]. Fairness can be measured at a community level as group fairness, which ensures that all types of users are treated equally, or as individual fairness which ensures that all individual users are treated equally. Many traditional network analysis techniques do not actively target fair results, and have been found to produce results that unfairly disadvantage certain communities [49]. Fairness has been investigated in a number of areas of SNA, and there are many comparative evaluations of anonymization methods, the fairness of anonymization algorithms has not been considered.

## 1.1 Contributions

In this thesis we compare the fairness with regards to community size of 19 different network anonymization methods. We select the set of 19 anonymization algorithms whose fairness we evaluate from survey papers, [20, 30, 68] which categorize graph anonymization methods. We focus on anonymizing networks where node attributes are not published, so we are concerned with preventing structure based reidentification. We focus on graph modification algorithms that publish a full graph, rather than node clusters or computed statistics. To evaluate the fairness of the anonymization algorithms, we use the anonymous graphs they produce, and compute privacy and utility metrics which we compare between minority and majority communities.

We look at the differences in anonymity and utility between communities on real world graphs with two non-equal communities. We partition graphs into communities based on node attributes. These node attributes are not used in the anonymization algorithms, and we assume that they would not be published with the graphs. We also investigate the relationship between fairness and the relative sizes of communities by using synthetic graphs with two communities and varying the relative community sizes. We look at fairness when graphs have more than two communities by using a set of synthetic graphs with multiple communities of different sizes.

## 1.2 Thesis Overview

In Chapter 2 we present some related works on the subject of network fairness and network anonymization. In Chapter 3 we give preliminary information. In Chapter 4 we present what anonymization algorithms we compare, and how we will evaluate them. In Chapter 5 we present the anonymity and utility scores of the different anonymization methods, and evaluate their fairness.

# Chapter 2

# Related Works

In this chapter we outline related work on graph anonymization, and fairness in SNA. We focus on how graph anonymization algorithms are distinguished, and how to measure their effectiveness. For fairness we consider what types of fairness have been proposed and how they are measured.

## 2.1 Fairness in SNA

Communities have structural differences, so SNA techniques perform differently on them, which can cause minorities to be unfairly disadvantaged. People are more likely to have connections to others who have the same attributes or interests. This homophily [51] principle means that people who are part of some minority group are likely to be less well connected to the majority group. This can cause minority groups to be unfairly disadvantaged in a number of areas. The random walks frequently used in node encodings are unlikely to visit nodes in racial minorities, causing these users to be underrepresented in downstream tasks such as link prediction [40]. Link recommendation algorithms might be less likely to recommend women to male dominated fields in professional connection sites [32], since links between male users are more common than links between female and male users. Information regarding HIV prevention is being less effectively distributed to queer people [56], since seed nodes for influence maximization are chosen only for overall utility. [49] is a comprehensive survey of the different types of fairness that have been investigated in SNA. The work distinguishes between individual and group fairness as types of fairness, where individual fairness considers differences in treatment across similar users. Group fairness considers inequality between different communities that exist in a network. The main fairness constraints outlined for group fairness are equity, equality, and statistical parity. Equity for groups is defined as different groups performing the same relative to their size. Equality for groups is defined as different groups receiving the same resources relative to group size [40]. Parity for groups is based on the variance of results over different groups [13]. In recent years, fairness has been explored in various downstream network analysis tasks, such as link prediction [47, 48], community detection [7, 11, 31], influence maximization [26, 41, 52], influence blocking [46], centrality ranking [21, 57], and opinion dynamics [55]. They also highlight areas of SNA that have yet to be investigated for fairness, including network anonymization.

## 2.2 Network anonymization

In this thesis, we focus on the algorithmic fairness of anonymization methods. There are a number of different ways that anonymization algorithms have been evaluated and categorized. The survey paper [30] classifies anonymization algorithms by what information they protect and by the anonymizing mechanisms. They assess a set of 46 graph anonymization algorithms, and note their anonymization mechanism as well as their merits and demerits with regard to anonymity, utility, complexity, and scalability. They give a detailed classification of different types of anonymity, graph utility, privacy threats, and anonymization operations. Another survey paper [68] provides ways to classify anonymization algorithms by the anonymizing mechanisms. For some algorithms they detail what information is protected by the anonymization, and what utility is preserved.

One common privacy technique is $k$-anonymity, which limits the ability of malicious actors to re-identify a node based on its structural attributes. The name $k$-anonymity comes from database anonymization, where it guarantees that for each entry, there are at least $k$ entries that are indistinguishable [9], so that there is only a $1/k$ chance of an adversary identifying an entry based on the attributes. In graphs, $k$-anonymity is based on some property of the nodes, such as attribute tags, degree, or neighborhood structure [3, 6, 10, 16, 24, 27, 65]. Another common anonymity approach is differential privacy (DP), which aims to hide the participation of particular nodes or edges in a network. This also originates from database anonymization and is designed for situations where queries are used to obtain statistics for data analysis. A database satisfies the $\epsilon$-differential privacy condition if the likelihood of receiving a query output from a database changed by one entry is within $e^{\epsilon}$, where $e$ is Euler's constant, of the likelihood of receiving that query output from the true database [45]. This is usually done by adding random noise to query results. In graph data, differential privacy is achieved when the existence in the original network of a node or edge cannot be determined from the anonymous graph [45]. Differential privacy-based anonymization algorithms randomly perturb aspects of the graph, such as the degree sequence or the graph edges [2, 19, 29, 39, 69]. There are also a number of other less popular privacy targets that algorithms may aim to achieve [59]. These may add uncertainty about edge endpoints, or shift important nodes [1, 14, 22, 23, 37, 62, 64].

The effectiveness of anonymization algorithms can be assessed using a variety of different anonymity metrics, which measure how well sensitive information is hidden. The paper [20] investigates several anonymization algorithms with regard to their resilience against different types of de-anonymization attacks. The paper also examines the information utility that is preserved by anonymization methods, and detail many ways to measure the graph utility. The paper [59] presents a compilation of different metrics to assess anonymity for different types of data, including social network data. The metrics are characterized based on what aspects of the graph or adversary they measure, and for each metric what aspects of privacy are considered, and the information required to compute it are specified.

The work [45] defines differential privacy for graph data using the DK series of a graph as the query function. The work proposes creating a differentially private graph by adding Laplace noise to the joint degree series, and publishing a graph constructed from the noised series. They provide a proof of $\epsilon$-differential privacy for their anonymization mechanism. The work [9] defines several $k$-anonymity measures on graph data, and compares their anonymization strength and their computational

efficiency.

## 2.3 Fairness in Network Anonymization

The combination of fairness and network anonymization remains unexplored. There are some algorithms that aim for a certain type of fairness. For instance [66] proposes a method for network anonymization that is fair to users' particular privacy settings, so that users who share more personal information receive better services than users who share less personal information. However there are no algorithms that focus on fairness across different user groupings.

The goal of this work is to start investigating fairness with regard to different communities in social network anonymization.

# Chapter 3

# Preliminaries

In this chapter we define the basic terminology that we use throughout the rest of the thesis. We define terms relevant to graph anonymization and graph theory in general.

## 3.1 Graph Properties

We define some of the graph terms and properties that we will make use of throughout this thesis. A network $G(V, E)$ is a set $V$ of $n$ nodes written $u, v \in V$, and a set $E$ of $m$ edges written $(u, v) \in E$ between those nodes. In this paper we refer interchangeably to networks and graphs. We use both vertices and nodes to refer to nodes, as well as edges or links to refer to edges. The degree of a node refers to the number of other nodes it is connected to. A directed graph is one where the edges have a direction, while in an undirected graph edges do not have a direction. In a weighted graph the edges have a weight value associated to them, to signify for instance length, or frequency of an edge. We focus on anonymization algorithms designed for simple graphs. Simple graphs are those that are unweighted and undirected, where each edge occurs only once, and nodes do not connect to themselves. We refer to the graphs before they are anonymized as the ground truth (GT) graphs, or as naively anonymized. Communities in a social network are groups of nodes created by some method. In this work we use communities that are created by grouping nodes based on their attribute values. Node attributes are values associated to a node, in social networks these might be a user's age, location, or gender.

The density of a graph is the number of edges relative to the maximum number of edges, computed as $\frac{2m}{n(n-1)}$. A sparse graph is one with few edges relative to the number of possible edges. The internal density of a community $c_i$ with $n_{c_i}$ nodes, is the number of actual intracommunity edges $m_{\text{intra comm}}$ relative to the number of possible intracommunity edges, computed as $\frac{m_{\text{intra comm}}}{0.5 n_{c_i}(n_{c_i}-1)}$ [5]. This is useful for seeing how strongly connected different communities are. The conductance of a community describes how well connected a community is to the remaining graph relative to its internal connections. This is computed as the number of intercommunity edges relative to the total number of edges of the community, $\frac{m_{\text{inter comm}}}{2m_{\text{intra comm}}+m_{\text{inter comm}}}$ [5].

We also define some of the mechanisms and graph properties that are used in the anonymization algorithms that we consider. The degree sequence of a graph is the list of the degrees of its nodes sorted from highest to lowest. The joint degree sequence

of a network is the sequence of degree pairs for each edge. This is also sorted from highest to lowest, first by maximal degree in the pair and then by minimal degree in a pair [45]. The $d$-neighborhood $N_d(u)$ of a node is all nodes that can be reached by taking at most $d$ edges from the node. A random walk starting from a node $u$ selects a random edge $(u, v)$ connected to $u$ and moves to its endpoint $v$. This is repeated for the length of the walk. The Jaccard Similarity of two nodes is computed as the Jaccard Similarity of their 1-neighborhoods. So $JS(u, v) = \frac{N_1(u) \cap N_1(v)}{N_1(u) \cup N_1(v)}$, that is, the number of neighbors they have in common, divided by their combined number of neighbors.

## 3.2 Graph Metrics

We analyze the utility of the anonymous networks by measuring how much the structure of the original graph was changed on a local and global scale. We assume that the more changes are made to the graph structure, the less utility is maintained.

The degree distribution of a graph is the frequency of the various degree values in a network. It provides insight into what type of network we are looking at, such as the whether the network has hub nodes, or is scale free. Centrality measures are useful to understand which nodes are important in the global structure of a network. The betweenness centrality (BC) measures the fraction of shortest paths that pass through a certain node. The highest BC nodes give an idea of the most important nodes when moving through the network. The distribution of the BC values of the nodes gives an idea of how regular the connections in the network are. If all nodes have very similar BC values, it indicates that all nodes are similarly connected, if there is a large disparity in BC values it indicates that some nodes are more strongly connected than others. Clustering measures are useful to understanding how densely connected a network is. The local clustering coefficient (LCC) is useful for the local structure of the graphs, in contrast to the global structure of the BC. The LCC of a node is the fraction of its neighbors that are connected to each other. This gives an idea of how densely connected the graph is. The distribution of local clustering coefficients gives insight into how densely and how evenly the graph is connected.

# Chapter 4

# Methodology

In this chapter we aim to present the details of the experiments that we performed. We outline information about the graph datasets that we use. We also provide the details of the experimental setup that we used.

## 4.1 Anonymization Algorithms

We chose the following anonymization algorithms to compare for our experiments.

### 4.1.1 ANO-NET

ANO-NET [10] is a computational framework for anonymizing, and measuring the anonymization of networks. The $k$-anonymization procedure can be applied to obtain $k$-anonymity for several different measures of anonymity, including the degree and neighborhood structure. The overall outline of the algorithm is as follows. For each of the edges a specified measure, such as the number of nodes affected by the removal, is computed to determine how much the edge affects anonymity. A specified number of the edges that have the highest impact are removed. This is repeated until a certain budget of removed edges is reached.

### 4.1.2 A New Method for Adding Noise Nodes

A New Method for Adding Noise Nodes [16], (ANNM) is proposed as an anonymization algorithm to achieve $k$-degree anonymity. The algorithm uses the degree sequence of the graph, and groups by degree such that each group has the same degree value. A target degree sequence is formed by combining small groups until all groups have at least size $k$. Edges are added for each group by iterating through the nodes by increasing betweenness centrality, and connecting them to noise nodes until the target degree is satisfied.

### 4.1.3 k-Degree Sequence Anonymization

The $k$-degree sequence anonymization algorithm [3] is proposed for $k$-degree anonymity. It works by creating a $k$-anonymous degree sequence, and modifying the edges to fit. The degree sequence is grouped by univariate microaggregation (UMGA) [54] into

groups of size at least $k$. The optimal target degrees for the groups are found using greedy optimization for the smallest absolute value of the sum of degree differences over all groups. The best solution is saved, and when no improvement is made the solution is returned. Edges are then added, removed, and moved to satisfy the target degrees.

### 4.1.4 Pseudo Node Anonymization

Pseudo Node or Dummy Node anonymization [6] is proposed for $k$-degree anonymity. The edges of the original graph are not changed, rather the graph is anonymized by adding new dummy nodes, and creating connections to them. The degree sequence of the graph is partitioned into sets of vertices of at least size $k$, and the maximal degree in each group is taken as the target degree. A number of dummy nodes are added to the graph, and are connected to the old nodes until all degree deficits are satisfied. The new edges are added by iterating through the new vertices in order, and through the original vertices in order of largest degree. This continues until all of the original vertices have their target degree. The dummy nodes are $k$-anonymized by connecting to each other.

### 4.1.5 k-Degree Anonymity with Vertex and Edge Modification

$k$-Degree Anonymity with Vertex and Edge Modification (KDVEM) [27] is proposed for $k$-degree anonymity. It partitions the degree sequence into groups with size at least $k$, using greedy minimization of the total degree deficit using the maximal degree in each partition group as the target degree. For each vertex, candidates for connection are selected based on distance and greedy modularity community. Edges are added in order between nodes and candidates until all degree deficiencies are satisfied. If there are then still nodes that are not sufficiently anonymous, dummy nodes are added and anonymized as in Pseudo Node Anonymization.

### 4.1.6 Two Step Anonymization

Two-Step anonymization [24] is proposed for $k$-degree anonymity. This algorithm partitions the degree sequence in the same manner as in Pseudo Node anonymization, but with regard to the total, rather than maximal degree deficiency. The algorithm then tries to reconstruct graph based on the target degree sequence. The graph is realized by choosing a random node, and connecting it to its neighbors highest degree first, and then to other nodes highest degree first. If the target sequence is not realizable, then the smallest degree value is increased using random noise, and the partition step is redone.

### 4.1.7 Differential Privacy with Field Theory

Differential Privacy with Field Theory (DPFT) [69] is proposed to achieve $\epsilon$-differential privacy. It creates a target degree sequence by adding Laplace random noise to each degree in the original sequence. To construct the anonymous graph, a set of candidate connections are created for each node. To create edges, nodes are selected with probability proportional to their target degree, and are probabilistically connected

to a candidate node. This is repeated until all candidate sets are empty. If the target degree sequence is not satisfied, additional edges are added between insufficiently connected nodes.

### 4.1.8 Edge Insertion Differential Privacy

Edge Insertion Differential Privacy (ENDP) [29] is proposed for $\epsilon$-differential privacy. The node list is sorted from lowest to highest degree. The anonymous graph is created by taking the same nodes as in the ground truth graph, initially all with degree $\theta$. From the lowest degree node edges are added connecting to its original neighbors, also beginning from the lowest degree nodes. Edges are only added if the current degree of both nodes is below a threshold value $\theta$.

### 4.1.9 Privacy Preservation Based on Clustering and Noise

Privacy Preservation Based on Clustering and Noise (PBCN) [19] is proposed for $\epsilon$-differential privacy. The algorithm does $k$-means clustering of the degree sequence into groups, and finds the average degree of each cluster. The groups are perturbed by adding Laplace noise to the cluster assignments of some nodes, and creating edges within the new groups. Then the degrees of each node are perturbed using Laplace noise to create a target degree sequence. This target sequence is reconstructed into a graph. As post processing, based on Laplace noise, either a number of noise nodes are added, and connected to true nodes, or a number of nodes with small degree are removed.

### 4.1.10 Subgraph Differential Privacy

Subgraph Differential Privacy [39] is proposed for $\epsilon$-differential privacy. A set of real and false edges are created, and each is assigned a weight and the graph is constructed using all edges below a certain weight. For each node, all of the real edges, as well as half of its degree value of false edges are taken from a breadth first search (BFS) of the node's neighborhood. This forms the set of edges. The edge weights are computed using common neighbors of the endpoints, and adding Laplace noise.

### 4.1.11 Optimization via Link Perturbation using Link Importance

Optimization via Link Perturbation using Link Importance (OLP) [62] has as the target of the algorithm is to obscure the link relationships between nodes, which they measure via the accuracy of structure prediction. The algorithm achieves this by removing the most regular edges in the graph. They compute the most regular edges by choosing a subset of the nodes at random, and from each of these nodes doing a random walk. The links in the walk are tracked, and each time an edge is used, its importance increases. The top 10% of edges by this metric are assumed to be the most regular, and are removed.

### 4.1.12 Edge Entropy

Edge entropy anonymization [64] is proposed to achieve high uncertainty about the edges to obscure the original graph, which they measure via edge entropy. In anonymiza-

tion, the 10% of the nodes with the highest betweenness centrality are selected, and those with large enough degree are classified as important nodes. From these important nodes probabilities are assigned to all node pairs in their neighborhoods. The anonymous graph is constructed by sampling edges using the assigned probabilities.

### 4.1.13 Link Perturbation

Link perturbation anonymization [37] is proposed with the goal of hiding node relationships. The algorithm creates a new set of edges by using a random walk procedure. For each vertex in the graph, a random walk is done from each of its original neighbors, and the vertex is with probability based on its degree connected to the endpoint of the random walk. If the proposed edge already exists in the graph, a new random walk procedure is done, until either a suitable endpoint is found, or the maximal number of attempts have been made.

### 4.1.14 Link Mirage

The link mirage algorithm [23] is proposed with the goal of hiding node relationships. The algorithm works by clustering the graph into several communities, and then perturbing the links within communities and between communities. The graph is clustered using the maximum modularity method. The edges within communities are perturbed using the same method as in Link Perturbation anonymization. To perturb the edges between communities, the inter-community edges are selected, and their endpoints changed.

### 4.1.15 Random Walk

The random walk anonymization algorithm [14] is proposed with the target of hiding the identity of hub nodes in a network. Nodes that are in the top 10% of degrees are designated as hub nodes. The algorithm does a random walk starting at the hub node, and the final node is designated the new hub node. For each hub, neighbors of the original hub are disconnected uniformly at random, and reconnected to the new hub node. This is repeated until the degree of the new hub node is the previous degree of the original hub node. After this shifting of hubs, the overall structure of the graph is perturbed as in Link Perturbation anonymization.

### 4.1.16 Upper Approximation based Anonymization

Upper Approximation based anonymization [22] is proposed with the goal of hiding node relationships. For each node the algorithm collects all nodes in the 2-neighborhood, and computes their Jaccard similarity based on their direct neighbors. The graph is reconstructed by connecting each node to all the nodes in the 2-neighborhood whose Jaccard similarity is greater than some threshold.

### 4.1.17 Friend in the Middle

Friend in the Middle (FiM) [1] anonymization is proposed for the purpose of protecting link privacy. To obscure links, a set of nodes from the network is selected uniformly at random, and a subset of the links in the network are redirected through these nodes. Several different methods of selecting the links to obscure are proposed, such

as selecting a proportion of nodes, or all nodes. The FiM nodes are selected in a fixed order, so that each FiM node is used equally.

## 4.2 Datasets

We ran the anonymization algorithms on a collection of 4 types of graphs. These are a mix of synthetic and real world networks. The real world networks are samples from Facebook and from Twitter. The synthetic networks that we use are a collection of 14 graphs simulated based on real data for health interventions [61] and a set of 25 graphs that we generated using the Homophily Barabasi Albert (HBA) method [50].

### 4.2.1 Facebook

One real world network is a graph of Facebook users where the "friendships" form the connections [35]. The graph is formed by the combination of 10 user's "friends", and their connections. This graph has 4039 nodes and 88234 edges. The community split here is formed by the gender attributes of the users. About 62% have gender attribute 0, and about 38% of nodes have gender attribute 1. This gives a distinct minority and majority group.

### 4.2.2 Twitter

The other real world networks that we use are sampled from a collection of twitter interactions [28], where the edges are based on "retweets". We refer to this set of graphs as Twitter graphs. Since "retweets" are directional, this is a directed network. The anonymization algorithms we consider are designed for graphs with undirected and no multiple edges. Because of this we regard each connection as bidirectional and flatten the network to an undirected graph, as well as considering only one edge for each node pair. After doing this the network has $367,188$ nodes and $816,869$ edges. The full network is too large for us to anonymize, so we used random sampling to get smaller graphs. We sampled the graph by adding each edge from the full graph with a probability of 0.015. Sampling based on the edges allowed us to get graphs without individual freestanding nodes. The communities in this graph are also formed by the gender of the users. About 80% of nodes have gender attribute value 0, and about 20% have gender 1.

### 4.2.3 Antelope Valley

The graphs simulated for health interventions are based on data from Antelope Valley, California [61]. We refer to this set of graphs as Antelope Valley or AV graphs. The graphs each have 500 nodes, between 915 and 967 edges, and geographic location, age, ethnicity, gender, and weight status as node properties. We chose to use ethnicity to determine the node communities. The graphs each split into 5 groups where the size split is approximately $50\%, 30\%, 10\%, 5\%, 5\%$.

### 4.2.4 HBA

We chose to use the HBA method to generate graphs, since it creates edges based on a homophily factor, and the community sizes can be given by hand [50]. The

| Graph | Nodes | | Relative Size | | Internal Density | | Conductance | |
|---|---|---|---|---|---|---|---|---|
| Community Label | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Facebook | 2507.0 | 1532.0 | 0.62 | 0.38 | 0.011 | 0.013 | 0.361 | 0.553 |
| Twitter | 21083.0 | 4804.4 | 0.81 | 0.19 | 0.000 | 0.000 | 0.153 | 0.756 |
| HBA 50/50 | 501.4 | 498.6 | 0.50 | 0.50 | 0.035 | 0.036 | 0.495 | 0.495 |
| HBA 60/40 | 605.4 | 394.6 | 0.61 | 0.39 | 0.033 | 0.039 | 0.407 | 0.574 |
| HBA 70/30 | 715.4 | 284.6 | 0.72 | 0.28 | 0.030 | 0.047 | 0.320 | 0.654 |
| HBA 80/20 | 806.8 | 193.2 | 0.81 | 0.19 | 0.030 | 0.049 | 0.220 | 0.750 |
| HBA 90/10 | 901.2 | 98.8 | 0.90 | 0.10 | 0.032 | 0.060 | 0.114 | 0.850 |

Table 4.1: Relative and absolute community sizes, internal density, and conductance of each of the communities in the Facebook, Twitter, and HBA datasets. The values are averages over the versions of the same graph.

| Graph | Nodes | | | | | Relative Size | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Community Label | 0 | 1 | 2 | 3 | 4 | 0 | 1 | 2 | 3 | 4 |
| Antelope Valley | 248.2 | 147.3 | 65.5 | 22.1 | 16.9 | 0.50 | 0.29 | 0.13 | 0.04 | 0.03 |
| | Internal Density | | | | | Conductance | | | | |
| Antelope Valley | 0.014 | 0.022 | 0.040 | 0.065 | 0.074 | 0.108 | 0.153 | 0.249 | 0.576 | 0.626 |

Table 4.2: Relative and absolute community sizes, internal density, and conductance of each of the communities of the average for the Antelope Valley networks.

graphs are generated by iteratively adding nodes and edges. Nodes are randomly assigned a community based on the relative community sizes, and edges are created within or between communities at random based on a specified homophile factor $h$. We create 25 graphs each with 1000 nodes and two communities. The homophily factor that we used was 0.2. We consider 5 differently minority and majority sizes, $(0.9, 0.1), (0.8, 0.2), (0.7, 0.3), (0.6, 0.4)$, and $(0.5, 0.5)$. For each of these we generate 5 graphs with the same parameters.

### 4.2.5 Community Designation

We use the number of nodes to designate communities as minority or majority, but the communities also differ in their number of edges and internal density. Table 4.1 and Table 4.2 show the number of nodes, internal density, and conductance for each of the communities in each of the datasets. These numbers are taken as the average over each type of graph.

In order to show that the communities of the attributed and generative graphs are comparable, we consider the conductance and internal density of the communities. We observe that the conductance of the real world network communities is similar to conductance in the HBA communities with similar relative size. This indicates that the minority communities in the HBA graphs have a similar level of connection to the majority communities as in real world networks. The AV graphs have a somewhat different range of conductance values, but we assume that they are informative since

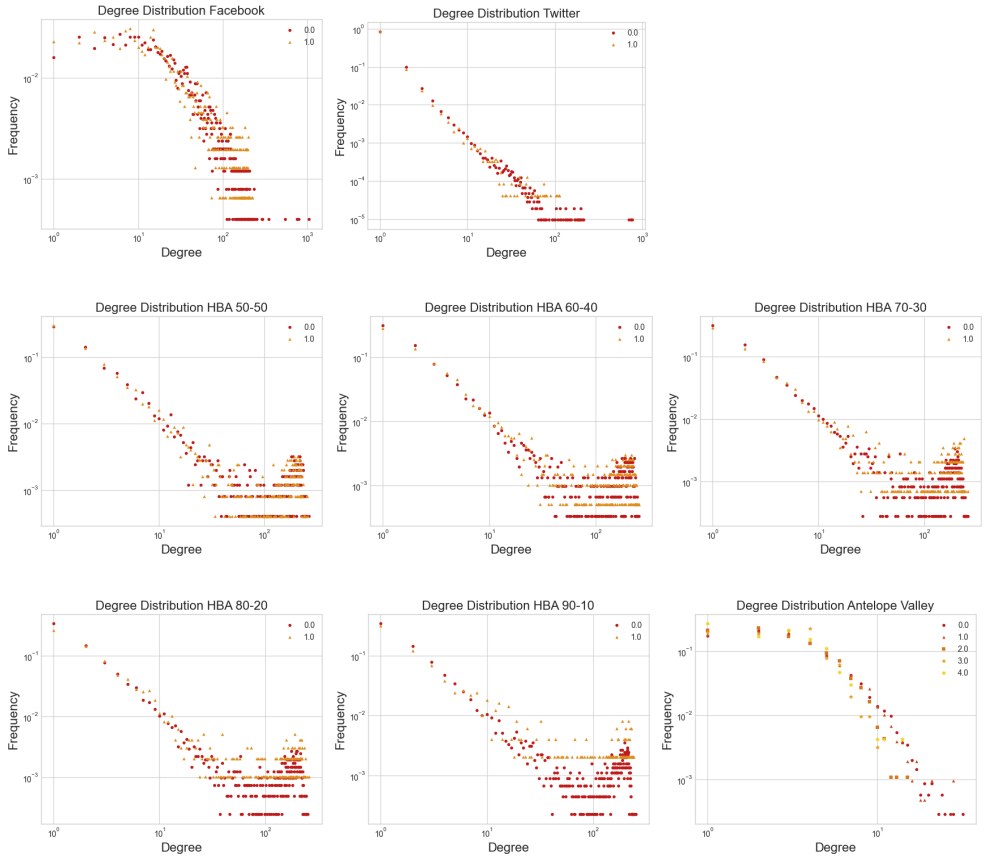



Figure 4.1: Degree distributions of the communities in each of the network datasets.

they are based on real medical data. We also note that the conductance is higher in minority communities, indicating they have more external connections than internal ones. In all the networks, the internal density is higher in smaller communities, and lower in larger communities, meaning that minority communities are more strongly connected than majorities. We note that the Twitter graph has very low internal density in both communities, being very sparsely connected. The Facebook graph also has slightly lower internal density than in the synthetic networks, however both the Facebook and Twitter networks are larger than the synthetic networks, which can contribute to the low connectivity.

### 4.2.6 Ground Truth Graph Structures

We consider the degree, LCC and BC distributions for the utility, so it is useful to know the distributions in the ground truth graphs, and how the distributions are different between the communities. This can help explain the behavior of the different anonymization algorithms in terms of fairness. Figure 4.1 shows the degree distribution of each of the communities for each of the datasets. The degree distribution for the Twitter network looks like that of a typical scale free network, with many

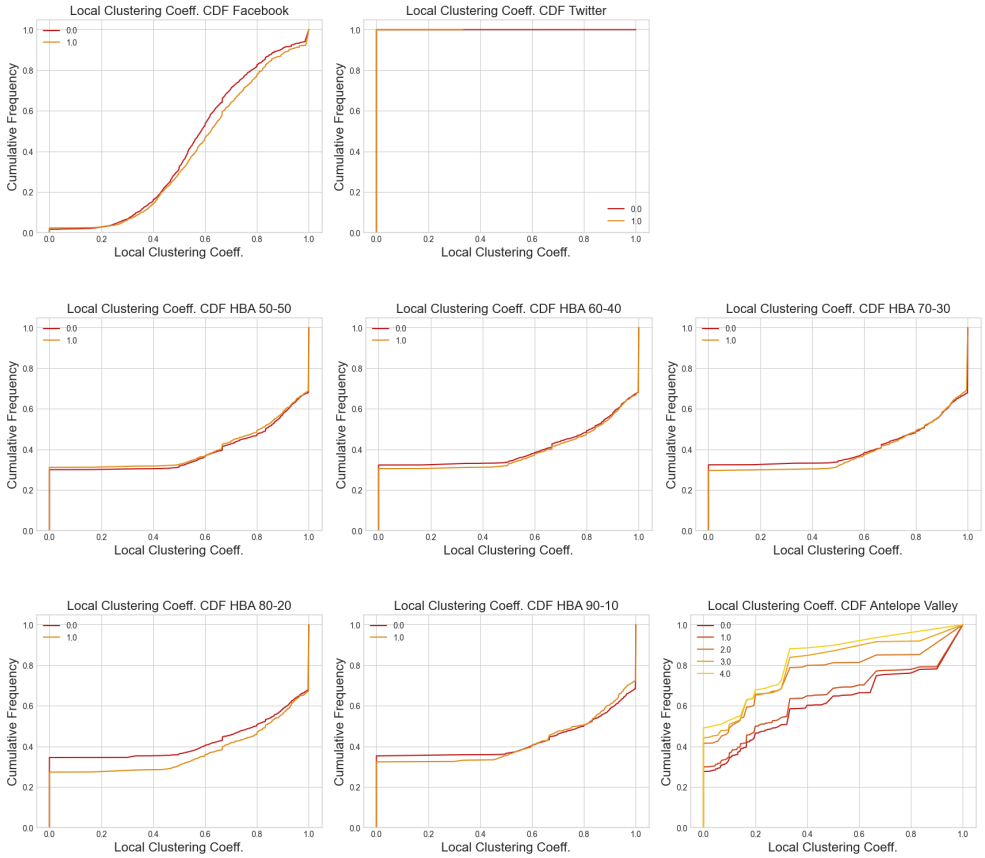



Figure 4.2: LCC distributions of the communities in each of the network datasets.

low degree nodes and few high degree nodes. The degree distribution of the Facebook network looks notably different, with there being a higher frequency of medium degree nodes. The degree distribution on the HBA graphs is similar to that of the Twitter network in the low degree nodes, but the HBA graphs all have a higher frequency of high degree nodes than of medium degree nodes. The degree distribution of the minority and majority communities are similar to each other, although the minority communities have a slightly higher relative incidence of high degree nodes. The degree distribution of the Antelope Valley graphs is quite similar to that of the Facebook network. The smaller communities on the Antelope Valley graphs have slightly fewer high degree nodes than the larger communities. Figure 4.2 shows the cumulative frequency distribution of the LCC values for each of the communities in the networks. In the Twitter network the minority and majority communities have a very similar LCC distribution, where the majority community has a few nodes with LCC 1 that the minority does not. On the Facebook network the minority community has slightly fewer nodes with a medium LCC value. The HBA graphs have relatively more low LCC nodes on the minority communities, although this is most pronounced on the 80/20 split rather than on the 90/10 split. The AV graphs follow a trend of having fewer high LCC nodes the smaller the communities. Figure 4.3 shows the

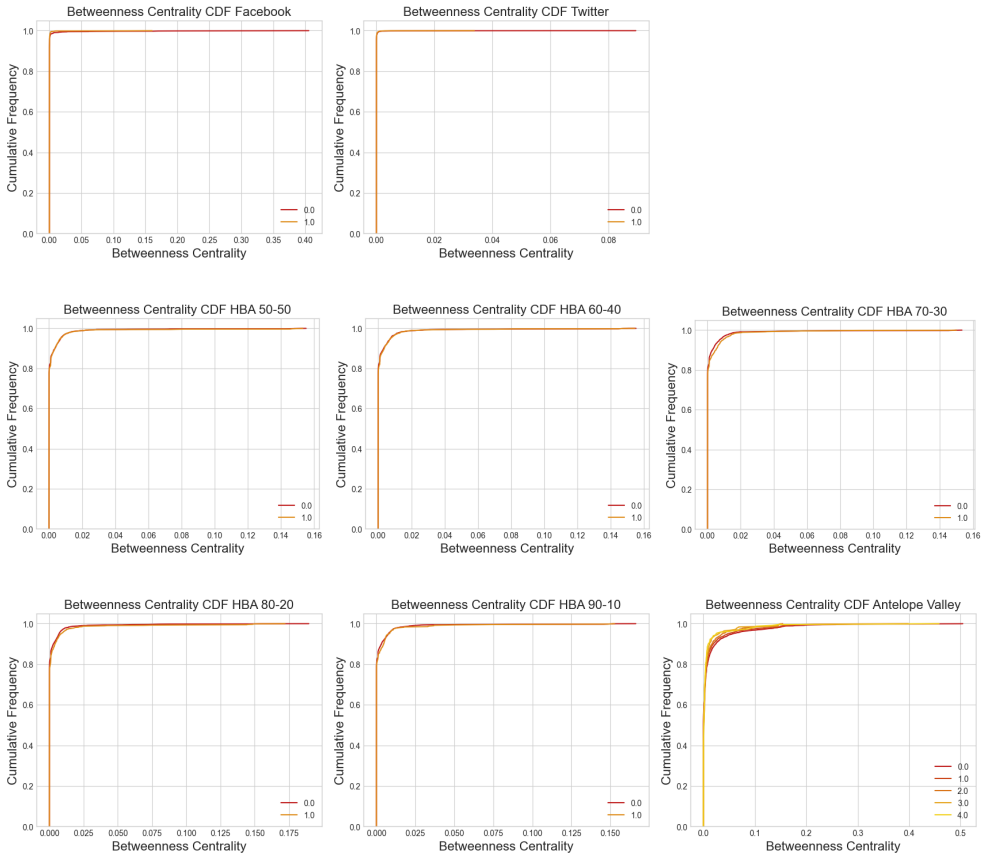



Figure 4.3: BC distributions of the communities in each of the network datasets.

cumulative frequency distribution of the BC values for each of the communities in the networks. For the betweenness centrality distribution, the minority communities have a lower maximal BC value than the majority communities. The AV graphs have more low BC nodes in the smaller communities than in the larger communities. The other graphs we considered have the same BC distribution between the minority and majority communities.

## 4.3 Utility Assessment

We want to consider the fairness with regard to both the anonymity and the utility of the anonymous graphs. For the anonymity we chose to use $k$-anonymity, and differential privacy to assess the privacy of each node.

For $k$-Anonymity we measured 5 different kinds of equivalence. Degree considers the degree of each node. Count considers the number of nodes and edges in the neighborhood of each node. Vertex refinement query (VRQ) considers all the degrees of the nodes in the neighborhood of each node. Isomorphism considers the full neighborhood structure of each node [9]. We compute the partitions based on these equivalence types, and modified it to output them for later analysis.

To measure differential privacy there was no pre-defined metric. We base our analysis on the definition of differential privacy for graph data from [45]. They define edge differential privacy on graphs by stating that an anonymization algorithm satisfies $\epsilon$-differential privacy if the following holds,

$$|\ln \frac{P[\tilde{DK}(G) = s]}{P[\tilde{DK}(G^*) = s]}| \leq \epsilon \tag{4.1}$$

where $\tilde{DK}$ is the privacy mechanism performed on the $dK$ series, $s$ is a possible output of $\tilde{DK}$, and $G$ and $G^*$ differ by at most one edge. In our computations the $dk$ sequence we use is the $d2$ sequence, that is, the joint degree sequence of the graph. We are primarily interested in measuring the node, rather than edge, differential privacy of a particular anonymous graph, so we consider the situation where $G$ and $G^*$ differ by at most one node. To estimate the likelihood of an anonymous graph originating from a certain ground truth graph, we use the number of changes in the joint degree sequence. We use this estimation to compute what we call a "Differential Privacy Score" for each node, as shown in eq. (4.2).

$$DP(v_0) = \max\left(\frac{d(D2(G), D2(G_A))}{d(D2(G \setminus v_0), D2(G_A))}, \frac{d(D2(G \setminus v_0), D2(G_A))}{d(D2(G), D2(G_A))}\right) \tag{4.2}$$

where $G$ is the ground truth graph, $G_A$ is the anonymous graph, $D2(G)$ is the joint degree series of $G$, and $d(a, b)$ is the distance between the joint degree series $a$ and $b$. The differential privacy score of a graph or community is the maximum score of its component nodes.

We use the following metrics to measure the privacy of the anonymous graphs.

- $k$-Degree anonymity
- $k$-Count (1-neighborhood) anonymity
- $k$-Count (2-neighborhood) anonymity

- $k$-VRQ (1-neighborhood) anonymity
- $k$-Isomorphism (1-neighborhood) anonymity
- Differential Privacy Score

## 4.4 Privacy Assessment

To assess the utility of the anonymous graphs, we compare the structural properties of the ground truth graph to those of the anonymous graphs, with the assumption that more different they are the less useful the anonymous graph. We consider the similarity of the anonymous graphs to the ground truth graph in several graph properties. We consider the distance to the ground truth graph via a Kolmogorov–Smirnov test (KS test) on the distribution of degrees, the distribution of local clustering coefficients, and the distribution of betweenness centrality values for each node. We compute the betweenness centrality using *nx.betweenness_centrality()* [15]. We compute the local clustering coefficients using *nx.clustering()* [15]. The KS test is a common way to measure the difference between distributions. It returns the maximum vertical distance between two cumulative distributions. We compute this using *scipy.stats.kstest()* [58]. We also consider the fraction of nodes that remain in the top 1% when ranked by betweenness centrality. To check the number of high BC nodes that remain the same, we get the 1% of nodes with the highest BC values for the GT and anonymous graphs and compute what fractions of the GT nodes remain. Since we consider primarily edge modification algorithms, the number of edge modifications that are made in the process of anonymizing is an intuitive way to assess the change in the graph. Which edges are added and removed from which nodes gives an idea of what nodes are most affected by anonymization. We compute the number of changed edges (CE) for each node by summing the number of edges added, and the number of edges removed during anonymization. We use $CE(v) = |N_G(v) \cup N_{G_A}(v)| - |N_G(v) \cap N_{G_A}(v)|$, where $N_G(v)$ is the 1-neighborhood of the graph $G$.

We use the following metrics to measure the utility of the anonymous graphs.

- KS difference in Degree distribution
- KS difference in BC distribution
- KS difference in LCC distribution
- Fraction of highest 10% BC value nodes unchanged
- Number of changed edges per node

## 4.5 Fairness Assessment

We assess the fairness between different community sizes by comparing the results of our utility and anonymity metrics on these communities. The communities in the anonymous graphs are defined by using the same node labels as in the GT graphs. We use equality fairness in our assessments, that is, we consider an algorithm fair in a certain metric if in the anonymous graph it creates all communities have the same score in that metric.

## 4.6 Experimental Setup

In this section we give technical information about the implementation of the anonymization algorithms and the experiments we performed.

### 4.6.1 Implementation of the Algorithms

In implementing the algorithms, we stayed as close as possible to the papers' descriptions of the methods. All of the anonymization algorithms output a single anonymous graph. All of the algorithms were implemented using Python 3.11.0[1]. The exception to this is ANO-NET [10], where we used the original C++code, and made modifications to save the graphs to file. We use the Networkx python package to create and store the graphs [15]. We implemented the metrics for measuring utility, and the differential privacy metric using Python 3.11.0, and NetworkX. We computed the $k$-anonymity partitions using the C++ code from ANO-NET [10].

Additional details for the implementation of some of the algorithms.

- ANO-NET
  - Edge selection method is AFF-U
  - Total Budget $b = \frac{|E|}{10}$
  - Edge cost recomputed every $\frac{b}{10} = \frac{|E|}{100}$ steps
  - Equivalence Set Types Degree, Count of vertices and edges in 1-Neighborhood, and degrees of each of the nodes in the 1-Neighborhood (Vertex Refinement Query or VRQ)
  - Equivalence set size $k = 10$
- ANNM
  - Equivalence set size $k = 10$
- $k$-Degree Sequence Anonymization
  - Equivalence set size $k = 10$
  - Max optimization iterations without change = 20
- Pseudo Node
  - Equivalence set size $k = 10$
- KDVEM
  - Equivalence set size $k = 10$
- Two Step Anonymization
  - Equivalence set size $k = 10$
  - Random Noise Range $[1, 5]$
- DPFT

[1] `https://github.com/FabianJennrich/Fairness-in-Network-Anonymisation`

  - Privacy Budget $\epsilon = 1$
  - Random Walk Length $t = 5$
  - Adjustment Factor $n = 1$
- ENDP
  - Maximal Degree Threshold $\theta = 80^{\text{th}}$ percentile of original degree values.
- PBCN
  - Number of means for clustering $k = 10$
  - Privacy budget $\epsilon = 0.1$
  - Degree threshold value for node removal $d_{\text{noise}} = 5^{\text{th}}$ percentile of original degree values.
- Subgraph DP
  - BFS tree depth limit $k = 5$
  - Privacy Budget $\epsilon = 5$
  - Ratio of False to True Edges in BFS tree $\beta = 0.5$
  - Edge weight Threshold $\theta = 0$
- OLP
  - Link importance measurement parameter $c = 0.1$
  - Fraction of nodes that start random walks $\alpha = 0.2$
  - Fraction of removed edges $\beta = 0.1$
- Link Perturbation
  - Random Walk Length $t = 5$
  - Maximum Failed Attempts per Edge $M = 10$
- Link Mirage
  - Random Walk Length $t = 5$
- Random Walk
  - Random Walk Length $t = 5$
- Upper Approximation
  - Jaccard Similarity Threshold $\delta = 0.2$
- FiM
  - Fraction of impacted Node Edges 0.2
  - Total number of FiM Nodes 50

### 4.6.2 Details on Running Testing

Many of the anonymization algorithms have a probabilistic element, so we do a series of five runs for each algorithm. We compute our metrics for each of the five anonymous graphs, and use the average of the metrics to compare the algorithms. The experiments were run on an Apple MacBook Pro with an Apple M2 Pro chip, and 32GB of memory, running macOS Sonoma 14.4.

# Chapter 5

# Results

In this section we present and discuss our results. We discuss the fairness of the anonymization algorithms on real world social networks, on synthetic networks with differently sized minority communities, and on synthetic networks with more than two communities.

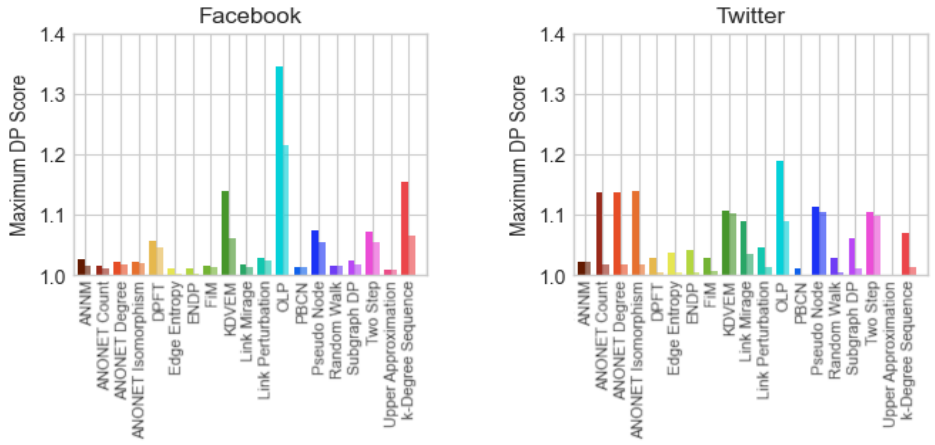
## 5.1 Fairness on Real World Social Networks

We investigate whether anonymization algorithms perform differently on the minority and majority communities created by node attributes on real world social networks. We test this by using the Facebook and Twitter graphs. We consider first the node privacy achieved by each algorithm, and then the utility.

Overall we observe in Figure 5.2 that minority communities tend to have better privacy scores while the majority communities tend to maintain better utility. We also note that on the utility metrics many of the algorithms perform inconsistently between the Facebook and Twitter graphs.

### 5.1.1 Privacy Fairness on Real World Social Networks

We see in Figure 5.1a that on the Facebook graph most of the algorithms have better differential privacy in the minority community, with the OLP, KDVEM, and $k$-Degree Sequence algorithms showing a particularly large difference between the communities. The Upper Approximation algorithm has very slightly better differential privacy on the majority community, and the PBCN algorithm shows no difference in the DP scores between majority and minority communities. In Figure 5.1b we see that on the Twitter graphs like on the Facebook graph, most of the algorithms produce better differential privacy on the minority community. For the Twitter graphs the ANONET, and OLP algorithms particularly show this. None of the algorithms favored the majority community, although for the ANNM algorithm the communities had the same DP score. The DP scores and fairness on DP scores is similar between the Facebook and Twitter graphs, although overall the DP scores are slightly better for the Twitter graphs. We note that the ANONET algorithms created a notably larger difference in DP score between communities on the Twitter graphs than on the Facebook graph. The OLP algorithm performed worst on both graphs. KDVEM created a strong preference for the minority community on the Facebook graph, but almost no difference




(a) Maximum DP score of the Facebook graph on the different anonymization algorithms.

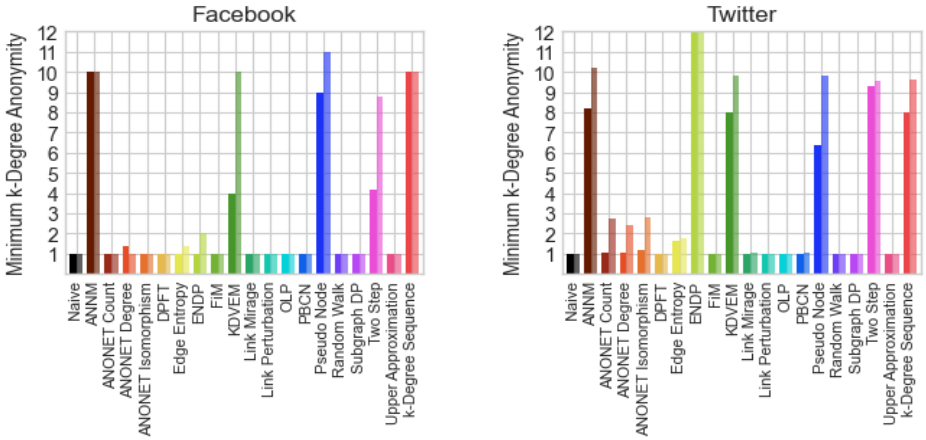
(b) Maximum DP score of the Twitter graphs on the different anonymization algorithms.

Figure 5.1: Differential Privacy scores after anonymization for the Facebook and Twitter datasets. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

between communities on the Twitter graphs. Most of the algorithms produce unfair results for DP, performing better on the minority community for both the Facebook and Twitter graphs.

In terms of $k$-degree anonymity, the naively-anonymized Facebook and Twitter networks both have only 1-degree anonymity in minority and majority communities. We see from Figure 5.2a that on the Facebook network most of the algorithms have only 1-degree anonymity, and there is no difference in the $k$-degree anonymity between minority and majority groups. The ANNM, and $k$-Degree Sequence algorithms achieve their goal of 10-degree anonymity in both communities. The ENDP, Edge Entropy, KDVEM, Pseudo Node, and Two Step algorithms favor the minority community. The ANONET Degree algorithm is the only one that favors the majority community. For the Twitter graphs, we see in Figure 5.2b that more of the algorithms have better degree anonymity in the minority group. The ANNM, ANONET Degree, ANONET Count, ANONET Isomorphism, $k$ Degree Sequence, KDVEM, Link Mirage, PBCN, Pseudo Node, Two Step algorithms have a better $k$-degree anonymity in minority community, while the other algorithms show no difference between the communities. None of the anonymization algorithms produced graphs with better $k$-degree anonymity in the majority.

For both the Facebook and Twitter graphs, the $k$-anonymity based algorithms have better degree anonymity than the other algorithms, but are also more likely to favor the minority community. This is likely because the minority community has fewer nodes, and since the algorithms anonymize all nodes simultaneously, or in a random order, it is more likely that nodes which failed to be anonymized are in the majority community than the minority. We note that the ENDP algorithm has about $k = 10700$ degree anonymity on the Twitter graphs, but only $k = 2$ degree anonymity on the Facebook graph. Most of the nodes in the Twitter graph have very low degree, so when the degree threshold of the ENDP algorithm cut off the less




(a) Minimum $k$-degree anonymity of the Facebook graph on the different anonymization algorithms.

(b) Minimum $k$-degree anonymity of the Twitter graphs on the different anonymization algorithms.

Figure 5.2: $k$-Degree Anonymity scores after anonymization for the Facebook and Twitter datasets. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

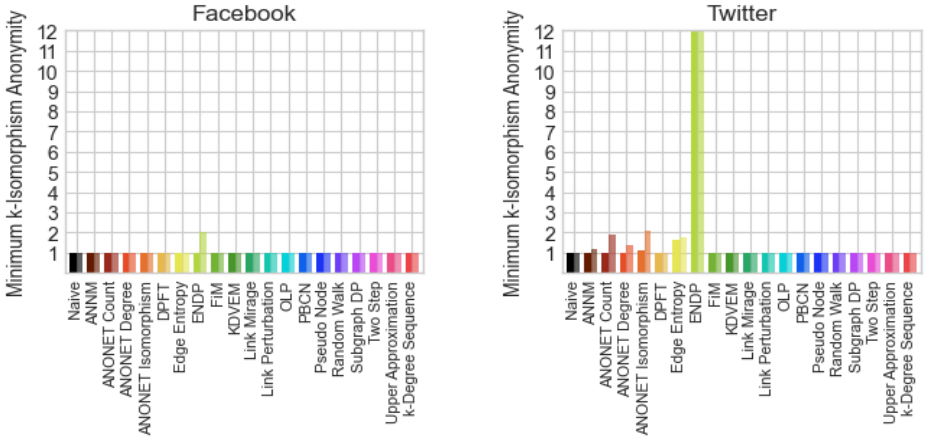
common higher degree values, all remaining degree values are very common. Only the Pseudo Node and Two Step algorithms favor the minority community for $k$-anonymity in both the Facebook and Twitter networks. The $k$-anonymity based algorithms aim to create 10-degree anonymity on all nodes, and there is unfairness if they fail to anonymize some nodes, which to be in the majority, since there are more nodes there. The DPFT, ENDP, FiM, Link Perturb, OLP, Random Walk, Upper Approximation, and Subgraph DP algorithms show no difference between the $k$-degree anonymity of communities on any graph.

For the other $k$-anonymity partitions that we computed, most of the algorithms only achieve 1-anonymity on both minority and majority communities. The highest anonymity is achieved on the $k$-Count with distance 1 and $k$-Isomorphism anonymity, where we see in Figure 5.3b that the ANNM, ANONET, Edge Entropy, and ENDP algorithms achieve better than 1-anonymity on the Twitter networks. The trend of minority communities performing better remains.

Overall for the privacy fairness on the real world networks, we observe that those algorithms that show unfairness on the Facebook and Twitter networks do so in a way that favors minority communities. Most of the anonymization algorithms showed unfairness either in the $k$-degree anonymity or in the differential privacy, but only the Pseudo Node and Two Step anonymization algorithms favoring the minority in both DP score and degree anonymity.

### 5.1.2 Utility Fairness on Real World Social Networks

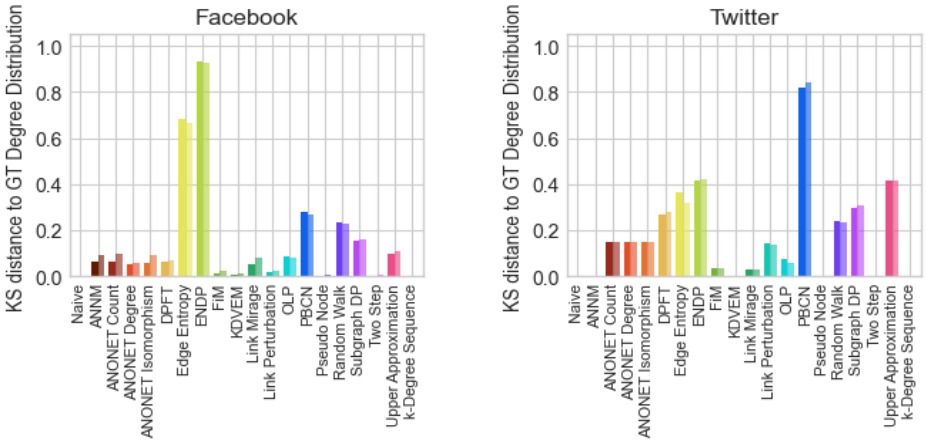
We see in Figure 5.4a that on the Facebook graph most of the anonymization algorithms favor the majority community in the degree distribution utility. The Edge Entropy, ENDP, OLP, PBCN, and Random Walk are the only algorithms to favor




(a) Minimum $k$-Isomorphism anonymity of the Facebook graph on the different anonymization algorithms.

(b) Minimum $k$-Isomorphism anonymity of the Twitter graphs on the different anonymization algorithms.

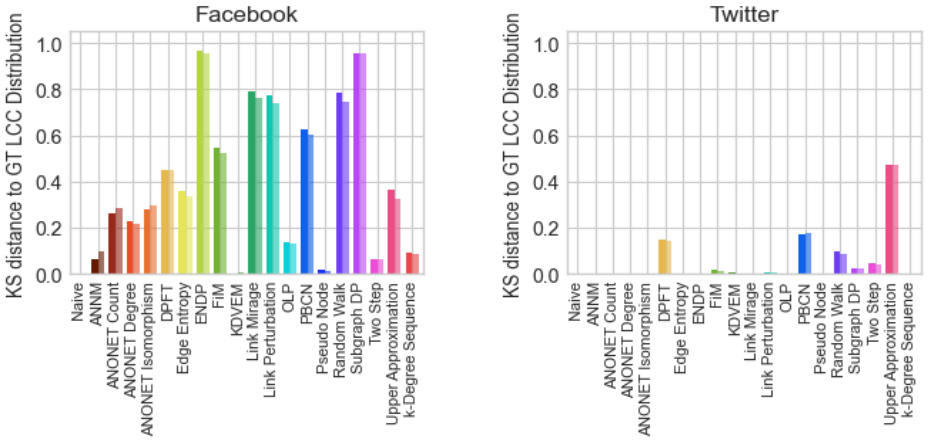
Figure 5.3: $k$-Degree Anonymity scores after anonymization for the Facebook and Twitter datasets. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.




(a) KS-statistic for distance to GT Degree Distribution of the Facebook graph on the different anonymization algorithms.

(b) KS-statistic for distance to GT Degree Distribution of the Twitter graphs on the different anonymization algorithms.

Figure 5.4: Changes in the Degree distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.




(a) KS-statistic for distance to GT LCC Distribution of the Facebook graph on the different anonymization algorithms.

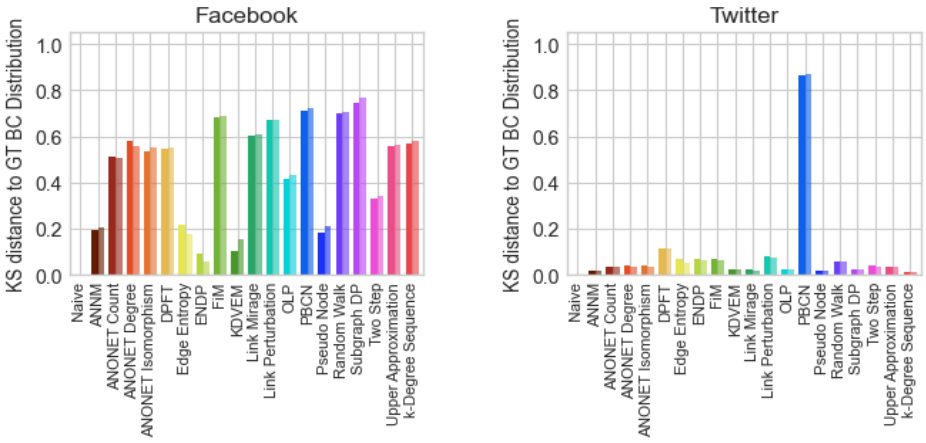
(b) KS-statistic for distance to GT LCC Distribution of the Twitter graphs on the different anonymization algorithms.

Figure 5.5: Changes in the LCC distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

the minority community. On the Twitter graphs, we see in Figure 5.4b that most of the algorithms have no difference in degree distribution utility between the minority and majority communities. The ANNM, Link Perturb, OLP, and Pseudo Node algorithms favor the minority community in the degree distribution utility. We note that the $k$-degree anonymity based algorithms tend to perform quite well, since many of them aim to minimize the changes to the degree sequence, and thus also in the degree distribution. The differences between the minority and majority community results are small where they exist.

For the LCC distribution, we see in Figure 5.5a that on the Facebook graph a number of the algorithms have better utility on the minority community. The ANNM, ANONET Count, ANONET Isomorphism, KDVEM, and Two Step algorithms favor the majority community, and the Subgraph DP algorithm shows no difference between minority and majority communities. On the Twitter graphs none of the algorithms consistently favored the majority community in terms of the LCC distribution utility. Most of the algorithms showed no difference in results between the minority and majority, and the DPFT, FiM, KDVEM, Random Walk, and Two Step algorithms favor the minority community. We observe that all except the Upper Approximation have very good LCC distribution utility on the Twitter graph, especially in comparison to the Facebook graph.

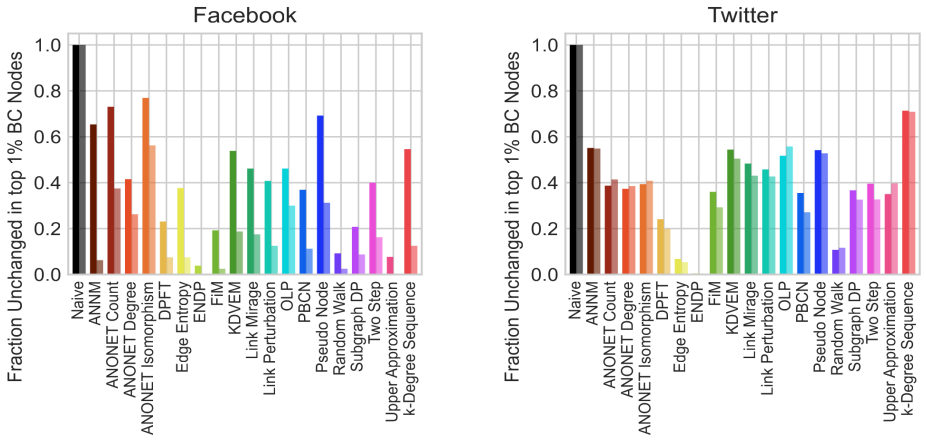
On the Facebook graph we see in Figure 5.6a that most algorithms favor the majority community in the BC distribution utility. The ANONET-Degree, ANONET-Count, Edge Entropy, and ENDP algorithms favor the minority community, and the Link Perturbation algorithm shows no difference between the minority and majority results. In the fraction of nodes in top 1% BC that remain unchanged after anonymization, shown in Figure 5.7a all algorithms favor the majority communities on the Facebook dataset. On the Twitter graphs we observe in Figure 5.6b that




(a) KS-statistic for distance to GT BC Distribution of the Facebook graph on the different anonymization algorithms.

(b) KS-statistic for distance to GT BC Distribution of the Twitter graphs on the different anonymization algorithms.

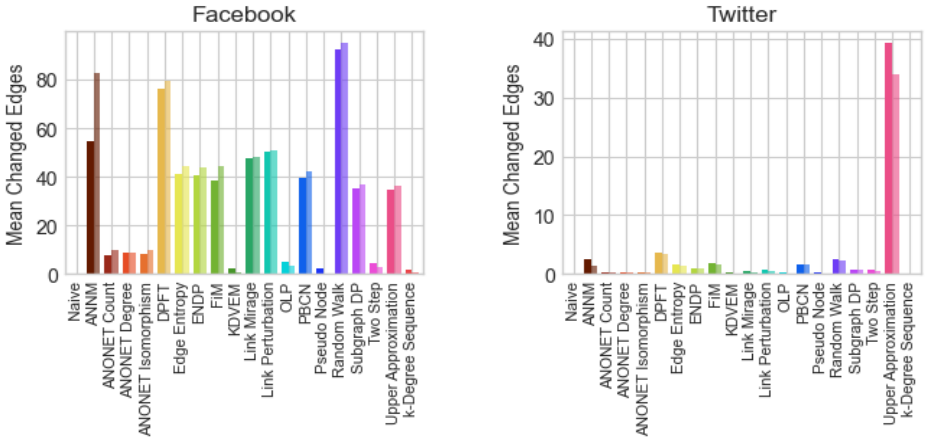
Figure 5.6: Changes in the BC distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.




(a) Number of top 1% BC nodes maintained for the Facebook graph with the different anonymization algorithms

(b) Number of top 1% BC nodes maintained for the Twitter graphs with the different anonymization algorithms

Figure 5.7: Unchanged Nodes in the top 1% highest BC value nodes during anonymization. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.




(a) Average number of changed edges per node for the Facebook graph with the different anonymization algorithms

(b) Average number of changed edges per node for the Twitter graphs with the different anonymization algorithms

Figure 5.8: Average number of edges per node that were changed during anonymization. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

the differences between the minority and majority BC distribution utility are small. Many algorithms have no difference at all. The ANONET Degree, ANONET Count, ANONET Isomorphism, ENDP, FiM, OLP, and Two Step algorithms favor the minority community, and only the PBCN algorithm favors the majority community. In the fraction of nodes in top 1% BC that remain unchanged after anonymization, many algorithms favor the majority community. The ANONET Count, ANONET Degree, ANONET Isomorphism, OLP, Random Walk, and Upper Approximation algorithms have better results in the minority community, and the ANNM and ENDP algorithms have no difference between minority and majority communities. We note that the unfairness on the Facebook graph for the fraction of nodes in top 1% BC that remain unchanged after anonymization, shown in Figure 5.7 is much stronger than on the Twitter graph.

The average number of changed edges per node is shown in Figure 5.8. On the Facebook graph most of the algorithms favored the majority community, with only the KDVEM, OLP, Pseudo Node, Two Step, and $k$-Degree Sequence algorithms favoring the minority community. On the Twitter graphs all of the algorithms performed slightly better on the minority community. We note that for the Twitter graphs the Upper Approximation algorithm performed much worse than the other algorithms.

Overall the differences between minority and majority utility is small, but largely favors the majority community. We observed that the $k$-degree anonymity algorithms tend to perform better on the distribution utilities than other algorithms. There do not appear to be trends with regard to more or less unfairness in different types of anonymization algorithms.

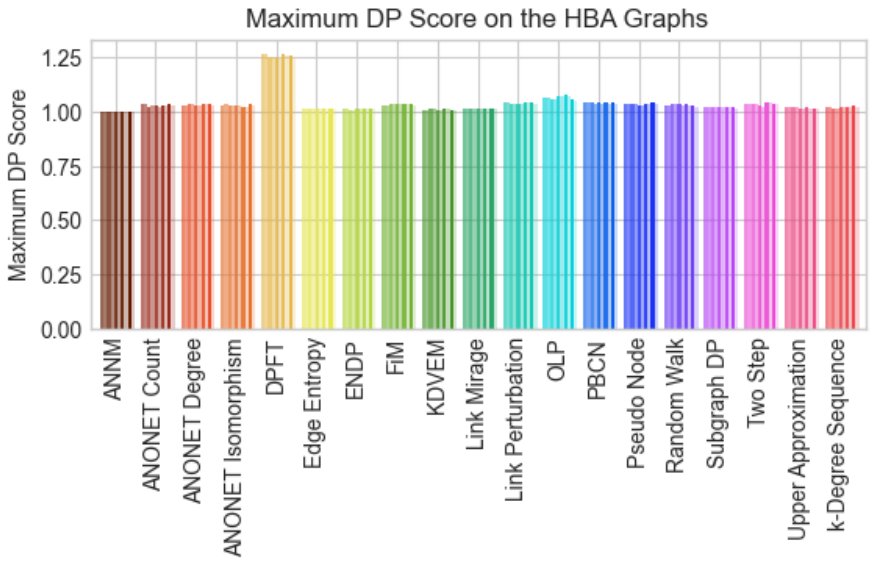



Figure 5.9: Differential Privacy scores after anonymization for the HBA datasets. For each algorithm the communities are in order of smallest to largest difference in community size, and the majority communities are colored darker than the minorities.

## 5.2 Fairness in Different Sized Minorities

We investigate how the fairness interacts with the relative size of the minority community. To test this we generate graphs using five different settings for the HBA generation. We generate 25 graphs each with 1000 nodes, and homophily factor 0.2. Each graph has two communities, where the majority community contains 50, 60, 70, 80, or 90 percent of the nodes. For each of these community splits we have five graphs. We consider the difference between the results for the majority and minority communities. We also consider how the results change as the community size changes, by plotting the communities of all graphs by size. We expect that if an algorithm favors minority or majority communities, that we see an increase in the difference between the results as the size difference between the communities increases. For algorithms that do not favor minority or majority communities we expect the difference between the majority and minority scores to be close to zero, or inconsistent with the scale of the size difference. If an algorithm performs differently based on the absolute size of communities, we expect to see a larger difference in the more extreme community splits, but we would also expect to see a correlation between the community size and metric scores.

### 5.2.1 Privacy Fairness over Different Community Sizes

We consider the DP score for each of the communities in Figure 5.9. For most algorithms we do not see the expected trend of larger DP scores for larger majority groups. We do observe that for most of the algorithms there is a larger difference in the DP score between the 90/10 communities than between the 50/50 split, and the minority communities tend to have a better DP score. For ANNM, ANONET Count,

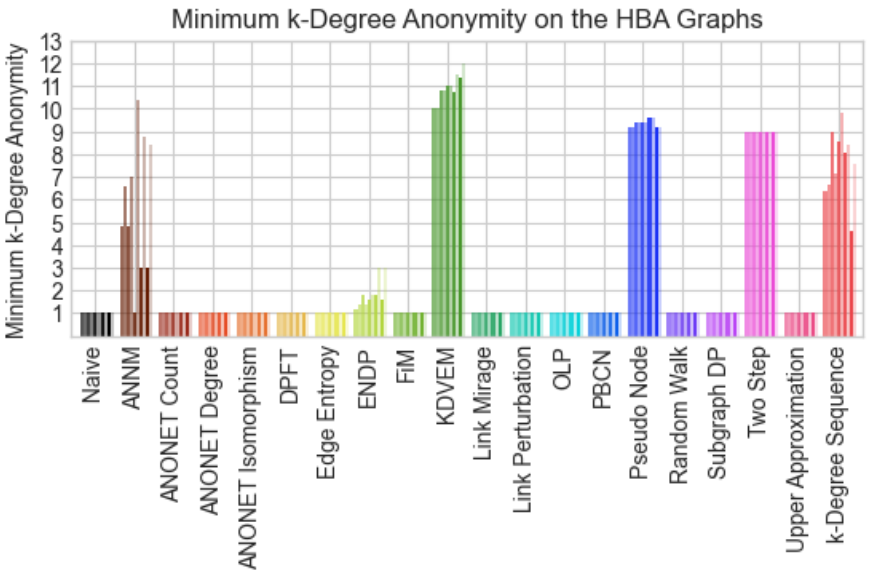



Figure 5.10: $k$-Degree Anonymity scores after anonymization for the HBA datasets. For each algorithm the communities are in order of smallest to largest difference in community size, and the majority communities are colored darker than the minorities.

ANONET Isomorphism, ENDP, Edge Entropy, Two Step, and Upper Approximation we observe no unfairness between the minority and majority communities. We observe that the differential privacy scores are more similar for majority and minority in the same graph, such as the 90% and 10%, than for communities of similar size in different graphs, such as the 90% and 80%. The utility appears to be better on minority groups, but the relative community size does not appear to influence the absolute DP score for any of the algorithms.

In Figure 5.10 we see that for most of the algorithms all of the considered communities have the same $k$-degree anonymity. The exceptions are the ANNM, ENDP, KDVEM, and $k$-Degree Sequence algorithms, which favor minority communities. In the ANNM, ENDP, and KDVEM algorithms we see the expected trend of increasing $k$-degree anonymity difference with increasing size difference, although the ANNM algorithm does have the largest difference between the 70% and 30% communities. For the $k$-Degree Sequence algorithm there does not appear to be a correlation to the difference in community sizes. For all of the other graph partitions we computed for $k$ anonymity, all algorithms produced only 1-anonymity on all communities.

### 5.2.2 Utility Fairness over Different Community Sizes

In the distribution utilities we see some of the expected trend with utility being related to the relative size of the communities. One example is the Subgraph DP algorithm in Figure 5.11 where the degree distribution utility is better as the relative community size decreases.

For the degree distribution utility in Figure 5.11 we see that most of the algorithms favor the majority over the minority communities. The exceptions are Link

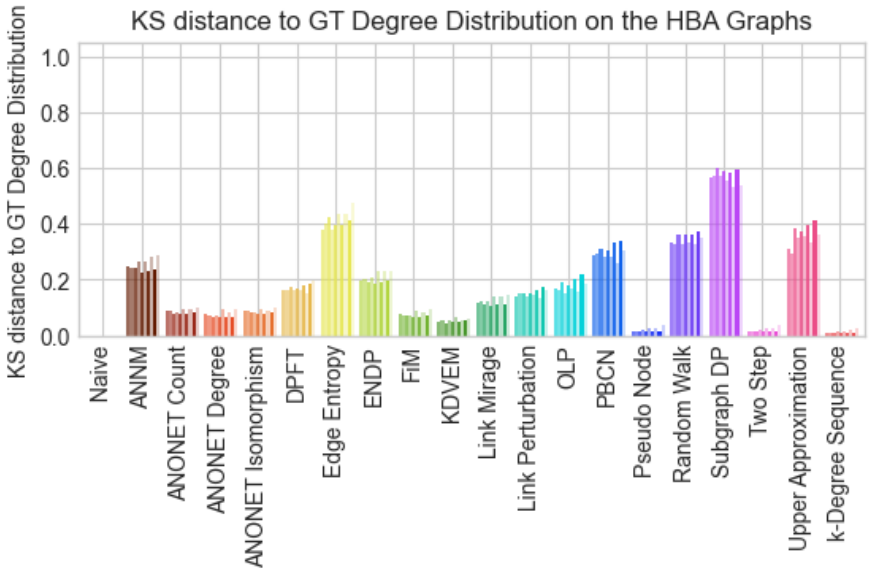



Figure 5.11: Changes in the Degree distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the communities are in order of smallest to largest difference in community size, and the majority communities are colored darker than the minorities.

Perturbation, OLP, PBCN, Random Walk, Subgraph DP, and Upper Approximation, which favor the minority. For many of the algorithms we do not see a trend in the absolute utility with relative community size, we only observe that there is a larger difference in utility between communities with larger size difference. The $k$-anonymity algorithms have very good degree distribution utility.

For the LCC distribution utility in Figure 5.12 we see similar trends as in the degree distribution utility. Most algorithms perform better on the majority communities, with only the Random Walk algorithm performing better on minorities. The OLP, PBCN, and Upper Approximation algorithms do not show clear favor towards either minority or majority groups. The Naive, KDVEM, Two Step, and $k$-Degree Sequence algorithms have the best LCC distribution utility, while the ENDP, Link Mirage and Subgraph DP algorithms have the worst.

For the BC distribution in Figure 5.13 we see for a lot of the algorithms less unfairness than for the degree and LCC distribution utilities. Most of the algorithms favor the majority communities in the BC distribution utility. The DPFT, Random Walk, and Subgraph DP algorithms do not favor either minority or majority, and Upper Approximation is the only algorithm to favor the minority. The Edge Entropy, ENDP, PBCN, Subgraph DP, and Upper Approximation algorithms have worse BC distribution utility than the other algorithms. We see the fraction of nodes in the top 1% of BC that remain unchanged in Figure 5.14. For a number of the algorithms there is a relatively large difference between the number of unchanged nodes in the two equally sized communities in the HBA 50/50 graphs, indicating that the difference between communities is unrelated to size. None of the algorithms show a clear favor towards either minority or majority communities.

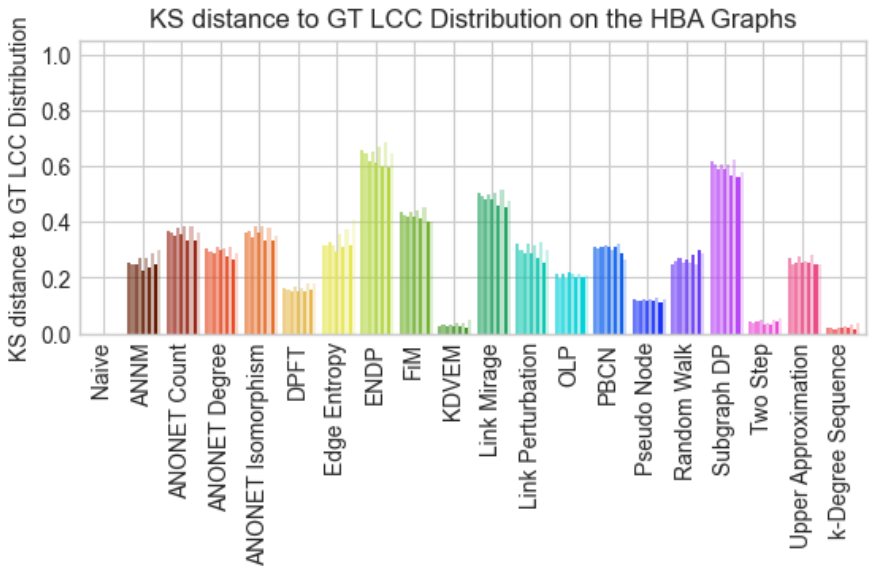



Figure 5.12: Changes in the LCC distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the communities are in order of smallest to largest difference in community size, and the majority communities are colored darker than the minorities.

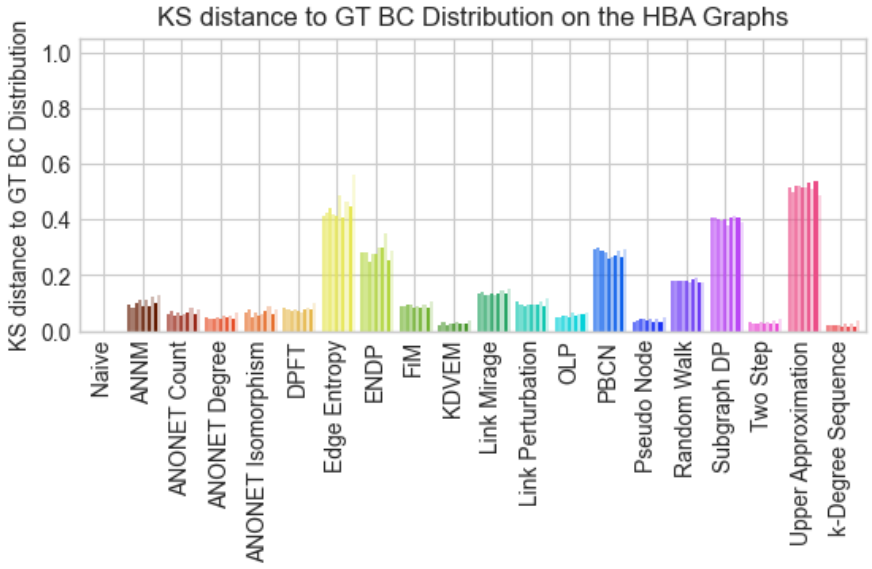



Figure 5.13: Changes in the BC distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the communities are in order of smallest to largest difference in community size, and the majority communities are colored darker than the minorities.

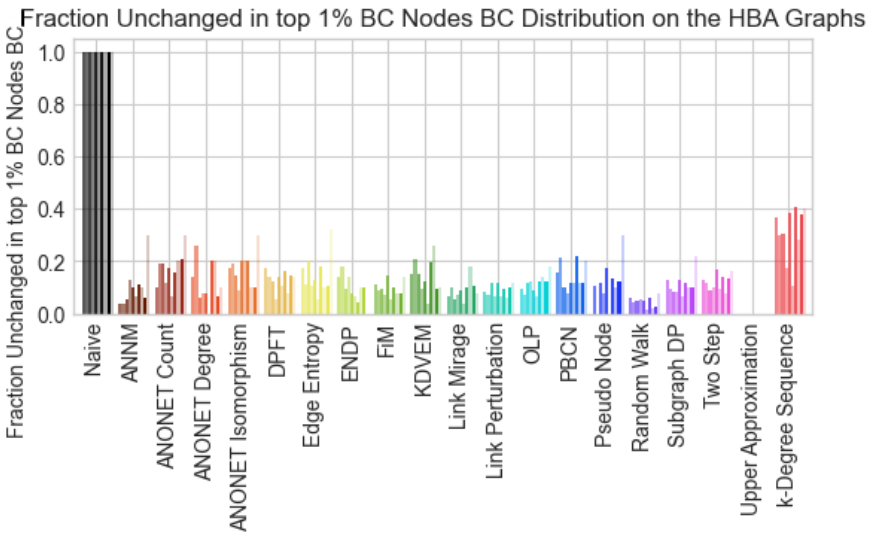



Figure 5.14: Changes in the top 1% highest BC value nodes during anonymization. For each algorithm the communities are in order of smallest to largest difference in community size, and the majority communities are colored darker than the minorities.

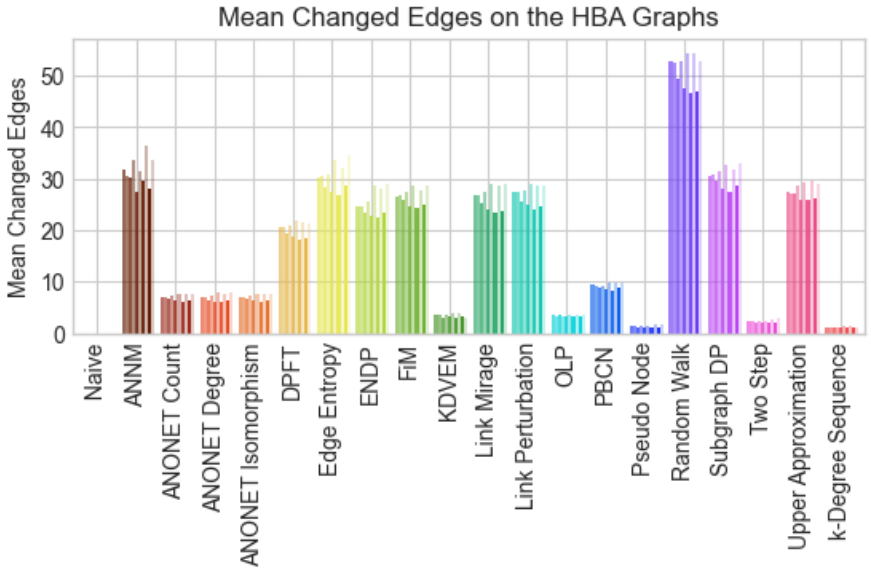



Figure 5.15: Average number of edges per node that were changed during anonymization. For each algorithm the communities are in order of smallest to largest difference in community size, and the majority communities are colored darker than the minorities.

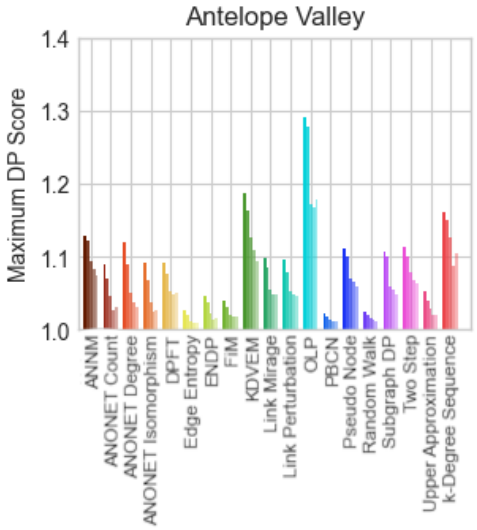



Figure 5.16: Differential Privacy scores after anonymization for the Antelope Valley dataset. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

For the average number of edges changed per node during anonymization in Figure 5.15 all of the algorithms are better on the majority communities. We note that the average number of changed edges per node increases with decreasing community size. Most of the $k$ anonymity algorithms have good overall utility in this metric.

## 5.3 Fairness with Multiple Minority Communities

The graphs we have considered so far are ones that have only two communities, with a clear distinction between minority and majority. There may be situations where there are more than two communities, so we must consider how fairness is affected when there is more than one minority group. We test this on the Antelope Valley graphs, where we partition based on ethnicity. This partitions the graphs into 5 communities, with a range of sizes. We consider the largest community, with around 50% of the nodes to be the majority community, and call the other communities minorities. We expect that if an algorithm favors minority communities, there is a trend from the smallest community performing the best and the largest community performing the worst. If an algorithm produces fair results we expect the results to be equal on all communities.

### 5.3.1 Privacy Fairness with Multiple Minority Communities

We observe from Figure 5.16 that the minority communities have better DP scores than the majority community for all of the anonymization algorithms, and that the DP scores are better the smaller the community. The difference between the majority and minority communities is least pronounced on the Edge Entropy, FiM, PBCN, and Random Walk algorithms, which also produce the best DP scores. We note that

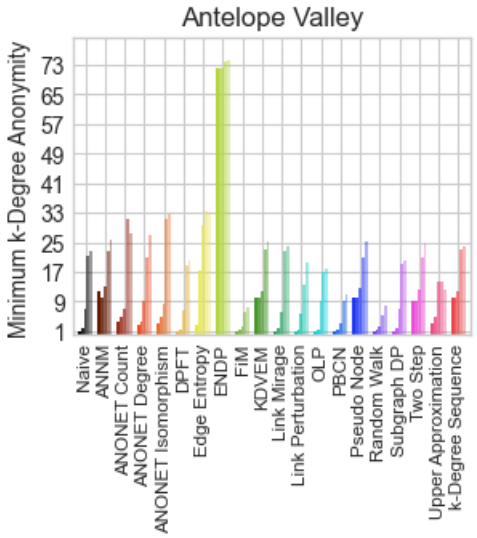



Figure 5.17: $k$-Degree Anonymity scores after anonymization for the Antelope Valley dataset. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

on ANNM, DPFT, FiM, Link Mirage, Link Perturbation, OLP, Pseudo Node, and Two Step the largest minority community performs similarly in the DP score to the majority, with a larger improvement to the smaller minorities.

We observe the same trend in Figure 5.17 where the $k$-degree anonymity is better on the minority communities across all anonymization algorithms. The other partitions that we computed for $k$-anonymity, seen in Figure 5.17 show the same trend. The AV graphs have higher $k$-anonymity values than the other graph datasets. For the other $k$-anonymity partitions that we computed, most of the algorithms achieve only low anonymity, and always favor the minority. The highest anonymity is achieved on the $k$-Count with distance 1 and $k$-Isomorphism anonymity, where we see in Figure 5.18 that the ANONET, Edge Entropy, and ENDP algorithms achieve the best anonymity.

We note that in terms of both DP and $k$-anonymity, the AV graphs show the strongest unfairness in privacy of the graph datasets we considered.

### 5.3.2 Utility Fairness with Multiple Minority Communities

For the degree distribution utility in Figure 5.19 we see that most algorithms perform better on the majority community, and on larger minorities. Only the ANNM, ENDP, KDVEM, and $k$-Degree Sequence algorithms perform better on minority communities than on the majority. The $k$-anonymity algorithms tend to maintain quite good degree distribution utility.

In the LCC distribution utility in Figure 5.20 we see that the ANNM, FiM, KDVEM, Link Mirage, Link Perturbation, Random Walk, and Subgraph DP algorithms perform better on minority communities, while the Edge Entropy, ENDP, PBCN, Pseudo Node, Two Step, Upper Approximation, and $k$-Degree Sequence algorithms

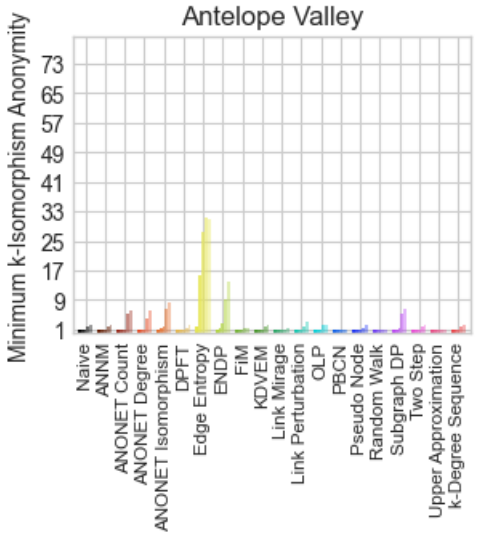



Figure 5.18: $k$-Isomorphism Anonymity scores after anonymization for the Antelope Valley dataset. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

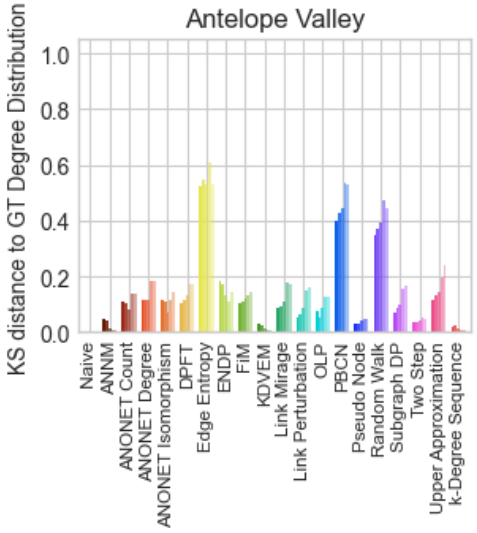



Figure 5.19: Changes in the Degree distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

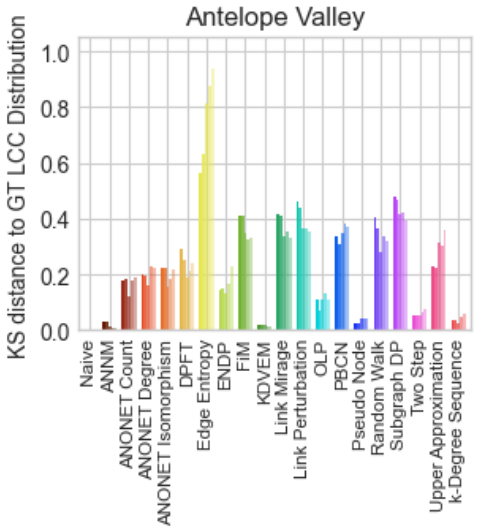



Figure 5.20: Changes in the LCC distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

perform better on the majority community. The remaining algorithms do not favor either the minority or majority communities. We also observe that a number of the algorithms achieve the best LCC distribution utility on the largest or second largest minority communities.

For the BC distribution utility seen in Figure 5.21, most of the algorithms favor the majority community. Only the DPFT, FiM, Link Perturbation, PBCN, Random Walk, and Subgraph DP algorithms favor the minority in the BC distribution utility. In the fraction of nodes in the top 1% BC values that remain unchanged after anonymization, we observe from Figure 5.22 that with exception of Naive, ANONET-Count, and ENDP anonymization, all algorithms performed worst on the second largest minority group, which has about 15% of nodes. We also see that many of the algorithms do not clearly favor the majority or minority communities, with the majority and smallest minority community performing similarly.

The average number of edges changed during anonymization per node is shown in Figure 5.23. Most of the algorithms have fewer changed nodes in the minority communities. Only the Upper Approximation algorithm has fewer changed edges in the majority community than in the minorities.

## 5.4 Results Summary

Based on the metrics that we use, all of the algorithms that we investigate produce better privacy in minority communities. We do not see consistent unfairness in the utility for most of the algorithms. With exception of the Random Walk algorithm which consistently maintains better LCC utility on minority communities, and the ANO-NET algorithms which maintain better degree distribution utility on majority communities, all algorithms that we investigate show different trends on the different

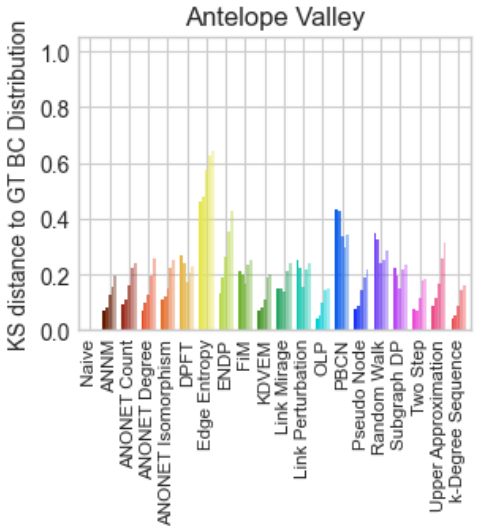



Figure 5.21: Changes in the BC distribution during anonymization as computed using the Kolmogorov–Smirnov distance. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

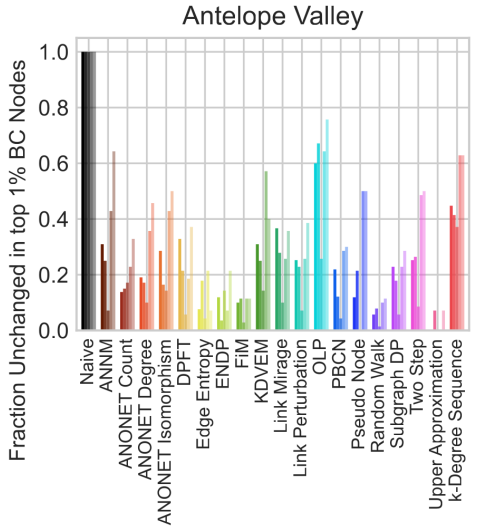



Figure 5.22: Changes in the top 1% highest BC value nodes during anonymization. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

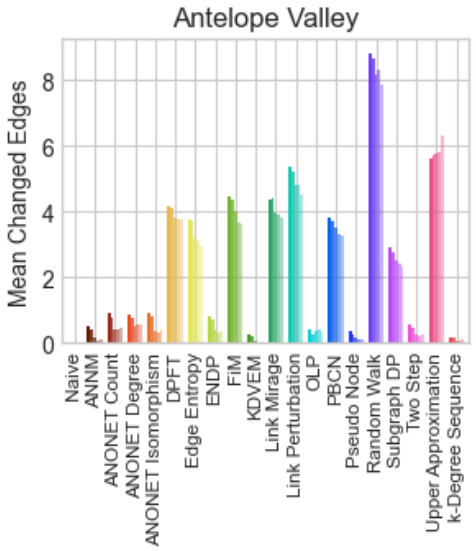



Figure 5.23: Average number of edges per node that were changed during anonymization. For each algorithm the majority community is depicted darker, and the minority community is depicted in a lighter color.

graphs that we test on.

We likely see better privacy in minority communities partially because they are smaller and have lower maximal degree values. Since $k$-anonymity and the DP score we use to measure privacy both use the worst case privacy, the most unique node is relevant. Since higher degree nodes are more likely to be unique, and require more resources to anonymize, they are more likely to have poor privacy scores after anonymization. We see in Table 5.1 that in each of the datasets we consider, the maximum degree of any node in the majority community is larger than the maximum degree of any node in the minority communities, making minority communities likely to have better privacy scores.

There is a connection between average degree value and fraction of unique nodes [43], implying that networks with higher average degree are likely to have more unique nodes. We do see that in most graphs the fraction of unique nodes is slightly higher for higher average degree. However, the fraction of unique nodes does not translate to $k$-anonymity or DP score, as we see that for the Facebook, Twitter, and HBA graphs the average degree is slightly larger in the minority community, despite the better privacy.

| Graph | Maximum Degree | | | | |
|---|---|---|---|---|---|
| Community Label | 0 | 1 | 2 | 3 | 4 |
| Facebook | 1045 | 224 | | | |
| Twitter | 771 | 113 | | | |
| HBA 50/50 | 266 | 258 | | | |
| HBA 60/40 | 252 | 254 | | | |
| HBA 70/30 | 258 | 257 | | | |
| HBA 80/20 | 251 | 258 | | | |
| HBA 90/10 | 253 | 241 | | | |
| Antelope Valley | 32 | 28 | 15 | 10 | 14 |

Table 5.1: Maximal degree values in each of the communities across the data sets. Community Label increases as Community Size decreases.

# Chapter 6

# Conclusion

Network anonymization hides information about nodes or edges when publishing a graph, with the goal of preventing malicious actors from accessing a user's personal information. A number of social network analysis and intervention methods show biased results for minority communities [49], and since network anonymization makes use of a number of the same methodologies, it may also introduce biases.

In this thesis we investigated the fairness with respect to community size for 19 different graph anonymization algorithms. We compute privacy and utility achieved by the anonymization algorithms on real world and synthetic networks. The real world networks are samples from the social networks Facebook and Twitter. One set of synthetic networks was generated using health data from Antelope Valley, while we created the other using the HBA model. We measured privacy using Differential Privacy and $k$-Anonymity metrics. We measured utility by considering the change in degree, local clustering coefficient, and betweenness centrality distributions created during anonymization, as well as the difference in the top ranked betweenness centrality nodes. We found that all of the algorithms that we investigated had better privacy results in minority communities. We also found that none of the algorithms we investigated consistently performed better or worse on minority communities over different graph types.

## 6.1 Future Research

We considered a fairly limited scope of anonymization algorithms, by focusing only on the anonymization of non-attributed simple graphs. Many social networks allow directional and multiple connections between users, and many users publish information about themselves which form node attributes. There are a range of anonymization algorithms that address privacy in directed networks [4, 18, 36], weighted edges [8, 25, 60], attribute safety [34, 38, 42], as well as sequential publishing of the same network over time [37, 44], and a number of types of anonymization we did not consider [12, 17, 33, 38, 53, 63, 67], and the fairness of these anonymization methods is worth investigating. Another direction of possible future work is investigating fairness in privacy with respect to particular de-anonymization procedures. We aim to gauge node safety using privacy metrics, but these may not be representative for all de-anonymization methods. In our experiments we found that frequently the fairness of anonymization algorithms was different on different graph types. Since real world

communities may differ in ways other than their size, for instance having different density, another area of future research may be investigating fairness of anonymization for communities that differ based on these properties. We also limited ourselves to partitioned communities, so investigating hierarchical or overlapping community structures may also be interesting.

# Bibliography


[1] F. Beato, M. Conti, and B. Preneel, "Friend in the middle (FiM): Tackling de-anonymization in social networks," in *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2013, pp. 279–284.

[2] P. Boldi, F. Bonchi, A. Gionis, and T. Tassa, "Injecting uncertainty in graphs for identity obfuscation," *Proc. VLDB Endow.*, vol. 5, no. 11, p. 1376–1387, 2012.

[3] J. Casas-Roma, J. Herrera-Joancomartí, and V. Torra, "k-degree anonymity and edge selection: improving data utility in large networks," *Knowledge and Information Systems*, vol. 50, pp. 447–474, 2016.

[4] J. Casas-Roma, J. Salas, F. D. Malliaros, and M. Vazirgiannis, "k-degree anonymity on directed networks," *Knowledge and Information Systems*, vol. 61, pp. 1743–1768, 2018.

[5] T. Chakraborty, A. Dalmia, A. Mukherjee, and N. Ganguly, "Metrics for community analysis: A survey," *ACM Computing Surveys*, vol. 50, no. 4, 2017.

[6] S. Chester, B. Kapron, G. Ramesh, G. Srivastava, A. Thomo, and V. Srinivasan, "Why waldo befriended the dummy? k-anonymization of social networks with pseudo-nodes," *Social Network Analysis and Mining*, vol. 3, pp. 381–399, 2012.

[7] F. Corriera, F. W. Takes, and A. Saxena, "Individual fairness in community detection: Quantitative measure and comparative evaluation," *arXiv preprint arXiv:2602.16326*, 2026.

[8] S. Das, Ö. Eğecioğlu, and A. El Abbadi, "Anonymizing weighted social network graphs," in *2010 IEEE 26th International Conference on Data Engineering (ICDE 2010)*, 2010, pp. 904–907.

[9] R. G. de Jong, M. P. J. van der Loo, and F. W. Takes, "A systematic comparison of measures for publishing k-anonymous social network data," *arXiv e-prints*, p. arXiv:2407.02290, 2024.

[10] R. G. de Jong, M. P. J. van der Loo, and F. W. Takes, "The anonymization problem in social networks," 2026.

[11] E. de Vink and A. Saxena, "Group fairness metrics for community detection methods in social networks," in *International Conference on Complex Networks and Their Applications*, 2024, pp. 43–56.

[12] T. Gao and F. Li, "Sharing social networks using a novel differentially private graph model," in *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, 2019, pp. 1–4.

[13] S. C. Geyik, S. Ambler, and K. Kenthapadi, "Fairness-aware ranking in search & recommendation systems with application to linkedin talent search," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '19. Association for Computing Machinery, 2019, p. 2221–2231.

[14] Y. Guo, Z. Liu, Y. Zeng, R. Wang, and J. Ma, "Preserving privacy for hubs and links in social networks," in *2018 International Conference on Networking and Network Applications (NaNA)*, 2018, pp. 263–269.

[15] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics, and function using NetworkX," in *Proceedings of the 7th Python in Science Conference*, G. Varoquaux, T. Vaught, and J. Millman, Eds., 2008, pp. 11–15.

[16] S. Hamzehzadeh and S. M. Mazinani, "ANNM: A new method for adding noise nodes which are used recently in anonymization methods in social networks," *Wireless Personal Communications*, vol. 107, pp. 1995–2017, 2019.

[17] M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis, "Resisting structural re-identification in anonymized social networks," *Proc. VLDB Endow.*, vol. 1, no. 1, p. 102–114, 2008.

[18] A.-T. Hoang, B. Carminati, and E. Ferrari, "Cluster-based anonymization of directed graphs," in *2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)*, 2019, pp. 91–100.

[19] H. Huang, D. Zhang, F. Xiao, K. Wang, J. Gu, and R. Wang, "Privacy-preserving approach PBCN in social network with differential privacy," *IEEE Transactions on Network and Service Management*, vol. 17, no. 2, pp. 931–945, 2020.

[20] S. Ji, P. Mittal, and R. Beyah, "Graph data anonymization, de-anonymization attacks, and de-anonymizability quantification: A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 1305–1326, 2017.

[21] M. Kumar, G. Dixit, and A. Saxena, "Fairness-sensitive pagerank approximation," 2026.

[22] S. Kumar and P. Kumar, "Upper approximation based privacy preserving in online social networks," *Expert Systems with Applications*, vol. 88, pp. 276–289, 2017.

[23] C. Liu and P. Mittal, "LinkMirage: Enabling privacy-preserving analytics on social relationships," in *Network and Distributed System Security Symposium*, 2016.

[24] K. Liu and E. Terzi, "Towards identity anonymization on graphs," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. Association for Computing Machinery, 2008, p. 93–106.

[25] Q. Liu, G. Wang, F. Li, S. Yang, and J. Wu, "Preserving privacy with probabilistic indistinguishability in weighted social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 5, p. 1417–1429, 2017.

[26] K. Ma, X. Xu, H. Yang, R. Cao, and L. Zhang, "Fair influence maximization in social networks: A community-based evolutionary algorithm," *IEEE Transactions on Emerging Topics in Computing*, vol. 13, no. 1, pp. 262–275, 2024.

[27] T. Ma, Y. Zhang, J. Cao, J. Shen, M. Tang, Y. Tian, A. Al-Dhelaan, and M. Al-Rodhaan, "KDVEM : a k-degree anonymity with vertex and edge modification algorithm," *Computing*, vol. 97, no. 12, pp. 1165–1184, 2015.

[28] M. Macedo and A. Saxena, "Gender differences in online communication: A case study of soccer," 2024.

[29] K. R. Macwan and S. J. Patel, "Node differential privacy in social graph degree publishing," *Procedia Computer Science*, vol. 143, pp. 786–793, 2018.

[30] A. Majeed and S. Lee, "Anonymization techniques for privacy preserving data publishing: A comprehensive survey," *IEEE Access*, vol. 9, pp. 8512–8545, 2021.

[31] K. Manolis and E. Pitoura, "Modularity-based fairness in community detection," in *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*, 2023, pp. 126–130.

[32] F. Masrour, T. Wilson, H. Yan, P.-N. Tan, and A. Esfahanian, "Bursting the filter bubble: Fairness-aware network link prediction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, pp. 841–848, 2020.

[33] S. Mauw, Y. Ramírez-Cruz, and R. Trujillo-Rasua, "Anonymising social graphs in the presence of active attackers," *Transactions on Data Privacy*, vol. 11, pp. 169–198, 2018.

[34] S. Mauw, Y. Ramírez-Cruz, and R. Trujillo-Rasua, "Conditional adjacency anonymity in social graphs under active attacks," *Knowl. Inf. Syst.*, vol. 61, no. 1, p. 485–511, 2019.

[35] J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *Proceedings of the 26th International Conference on Neural Information Processing Systems*, ser. NIPS'12, vol. 1. Curran Associates Inc., 2012, p. 539–547.

[36] A. Milani Fard and K. Wang, "Neighborhood randomization for link privacy in social network analysis," *World Wide Web*, vol. 18, no. 1, p. 9–32, 2015.

[37] P. Mittal, C. Papamanthou, and D. X. Song, "Preserving link privacy in social network based systems," 2012.

[38] D. Mohapatra and M. R. Patra, "Anonymization of attributed social graph using anatomy based clustering," *Multimedia Tools Appl.*, vol. 78, no. 18, p. 25455–25486, 2019.

[39] B. P. Nguyen, H. Ngo, J. Kim, and J. Kim, "Publishing graph data with subgraph differential privacy," in *Information Security Applications*, H.-w. Kim and D. Choi, Eds. Springer International Publishing, 2016, pp. 134–145.

[40] T. Rahman, B. Surma, M. Backes, and Y. Zhang, "Fairwalk: Towards fair graph embedding," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 2019, pp. 3289–3295.

[41] A. Rahmattalabi, S. Jabbari, H. Lakkaraju, P. Vayanos, M. Izenberg, R. Brown, E. Rice, and M. Tambe, "Fair influence maximization: A welfare optimization approach," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 13, 2021, pp. 11 630–11 638.

[42] A. K. Rizi, M. N. Dehkordi, and N. N. bakhsh, "SNI: Supervised anonymization technique to publish social networks having multiple sensitive labels," *Secur. Commun. Networks*, vol. 2019, pp. 8 171 263:1–8 171 263:23, 2019.

[43] D. Romanini, S. Lehmann, and M. Kivelä, "Privacy and uniqueness of neighborhoods in social networks," 2020.

[44] B. Safia and C. Yacine, "Privacy preservation in social networks sequential publishing," in *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, 2018, pp. 732–739.

[45] A. Sala, X. Zhao, C. Wilson, H. Zheng, and B. Y. Zhao, "Sharing graphs using differentially private graph models," in *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, ser. IMC '11. Association for Computing Machinery, 2011, p. 81–98.

[46] A. Saxena, C. G. Bierbooms, and M. Pechenizkiy, "Fairness-aware fake news mitigation using counter information propagation," *Applied Intelligence*, vol. 53, no. 22, pp. 27 483–27 504, 2023.

[47] A. Saxena, G. Fletcher, and M. Pechenizkiy, "HM-EIICT: Fairness-aware link prediction in complex networks using community information," *Journal of Combinatorial Optimization*, vol. 44, no. 4, pp. 2853–2870, 2022.

[48] ——, "NodeSim: node similarity based network embedding for diverse link prediction," *EPJ Data Science*, vol. 11, no. 1, p. 24, 2022.

[49] ——, "FairSNA: Algorithmic fairness in social network analysis," *ACM Computing Surveys*, vol. 56, no. 8, 2024.

[50] A. Saxena, C. Gutiérrez Bierbooms, and M. Pechenizkiy, "Fairness-aware fake news mitigation using counter information propagation," *Applied Intelligence*, vol. 53, no. 22, p. 27483–27504, 2023.

[51] A. Saxena, G. Kumar, and C. Meena, "Homophily in complex networks: Measures, models, and applications," *arXiv preprint arXiv:2509.18289*, 2025.

[52] A. Saxena, H. K. Yadav, B. Rutten, and S. S. Jha, "Dq4fairim: Fairness-aware influence maximization using deep reinforcement learning," *arXiv preprint arXiv:2512.00545*, 2025.

[53] M. Siddula, Y. Li, X. Cheng, Z. Tian, and Z. Cai, "Anonymization in online social networks based on enhanced equi-cardinal clustering," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 4, pp. 809–820, 2019.

[54] F. I. Stamm and M. T. Schaub, "Faster optimal univariate microgaggregation," 2024.

[55] S. Stepień, M. Janik, M. Nurek, A. Saxena, and R. Michalski, "Fairness in opinion dynamics," *arXiv preprint arXiv:2601.03859*, 2026.

[56] A. Tsang, B. Wilder, E. Rice, M. Tambe, and Y. Zick, "Group-fairness in influence maximization," 2019.

[57] S. Tsioutsiouliklis, E. Pitoura, P. Tsaparas, I. Kleftakis, and N. Mamoulis, "Fairness-aware pagerank," in *Proceedings of the Web Conference 2021*, 2021, pp. 3815–3826.

[58] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[59] I. Wagner and D. Eckhoff, "Technical privacy metrics: A systematic survey," *ACM Computing Surveys*, vol. 51, no. 3, p. 1–38, 2018.

[60] D. Wang and S. Long, "Boosting the accuracy of differentially private in weighted social networks," *Multimedia Tools and Applications*, vol. 78, pp. 34 801–34 817, 2019.

[61] B. Wilder, H. C. Ou, K. de la Haye, and M. Tambe, "Optimizing network structure for preventative health," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, ser. AAMAS '18. International Foundation for Autonomous Agents and Multiagent Systems, 2018, p. 841–849.

[62] T. Wu, G. Ming, X. Xian, W. Wang, S. Qiao, and G. Xu, "Structural predictability optimization against inference attacks in data publishing," *IEEE Access*, vol. 7, pp. 92 119–92 136, 2019.

[63] Q. Xiao, R. Chen, and K.-L. Tan, "Differentially private network data release via structural inference," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14. Association for Computing Machinery, 2014, p. 911–920.

[64] J. Yan, L. Zhang, Y. Tian, G. Wen, and J. Hu, "An uncertain graph approach for preserving privacy in social networks based on important nodes," in *2018 International Conference on Networking and Network Applications (NaNA)*, 2018, pp. 107–111.

[65] A. A. Yarifard and S. Dehnvai, "Link perturbation in social network analysis through neighborhood randomization," *2015 4th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*, pp. 1–7, 2015.

[66] X. Zheng, G. Luo, and Z. Cai, "A fair mechanism for private data publication in online social networks," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 880–891, 2020.

[67] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in *2008 IEEE 24th International Conference on Data Engineering*, 2008, pp. 506–515.

[68] B. Zhou, J. Pei, and W. Luk, "A brief survey on anonymization techniques for privacy preserving publishing of social network data," *SIGKDD Explorations Newsletter*, vol. 10, no. 2, p. 12–22, 2008.

[69] H. Zhu, X. Zuo, and M. Xie, "DP-FT: A differential privacy graph generation with field theory for social network data release," *IEEE Access*, vol. 7, pp. 164 304–164 319, 2019.