# Anonymizing Graphs: Measuring Quality for Clustering

Jordi Casas-Roma, Jordi Herrera-Joancomartí, and Vicenç Torra

Universitat Oberta de Catalunya (UOC), Barcelona, Spain. E-mail: jcasasr@uoc.edu
Universitat Autònoma de Barcelona (UAB), Bellaterra, Spain. E-mail: jherrera@deic.uab.cat
Artificial Intelligence Research Institute (IIIA), Spanish National Research Council (CSIC),
Bellaterra, Spain. E-mail: vtorra@iiia.csic.es

**Abstract.** Anonymization of graph-based data is a problem which has been widely studied last years and several anonymization methods have been developed. Information loss measures have been carried out to evaluate the noise introduced in the anonymized data. Generic information loss measures ignore the intended anonymized data use. When data has to be released to third-parties, and there is no control on what kind of analyses users could do, these measures are the standard ones. In this paper we study different generic information loss measures for graphs comparing such measures to the cluster-specific ones. We want to evaluate whether the generic information loss measures are indicative of the usefulness of the data for subsequent data mining processes.

**Keywords:** Privacy, Networks, Data mining, Mining methods and algorithms, Quality and Metrics, Semi-structured Data and XML

## 1 Introduction

Currently, the data mining processes require large amounts of data, which often contain personal and private information of users and individuals. Although basic processes are performed on data anonymization, such as removing names or other key identifiers, remaining information can still be sensitive, and useful for an attacker to re-identify users and individuals. To solve this problem, methods which introduce noise to the original data have been developed in order to hinder the subsequent processes of re-identification. However, the noise introduced by the anonymization processes may affect the data, reducing its usefulness in subsequent processes of data mining. It is necessary to keep the main properties of the data to ensure the data mining process is not altered by the anonymization process.

The anonymization processes should allow the analysis performed into the anonymized data lead to results as equal as possible to the ones obtained when applying the same analysis to the original data. Nevertheless, data modification is in contradiction with data utility. The larger data modification, the less data utility. Thus, a good anonymization method hinders the re-identification process while causing minimal distortion in the data.

Owing to what we have mentioned in the previous paragraph, several measures have been designed to evaluate the goodness of the anonymization methods. Generic information loss measures evaluate in what extent the analysis on anonymized data differs from the original data. Each measure focuses on a particular property of the data. We assume that if these metrics show little variation between original and anonymized data, then the subsequent data mining processes will also show little variation between original and anonymized data. However, the behaviour of anonymized data in the subsequent data mining processes may not coincide with the expected results. Since evaluating the distortion introduced in the graph is not enough, it is necessary to assess the noise introduced in the subsequent data mining processes. No analysis has been made to evaluate whether these measures are suitable to accommodate the information loss when data are used to specific purposes. In our work we consider the case of clustering-specific processes.

### 1.1 Our contributions

In this paper we compare some generic information loss measures to clustering-specific ones on graph formatted data. We evaluate whether such generic information loss measures predict the divergence between the clusters obtained from the original data and the clusters obtained from the anonymized data, correctly. We offer the following results:

- We analyse the behaviour of some generic and clustering-specific information loss measures and demonstrate that some measures behave in similar way independently of the dataset where they are applied. On the contrary, others present a behaviour subordinated to the applied dataset.
- We demonstrate that some generic information loss measures are strongly correlated to clustering-specific measures; while others present moderate correlation and few of them do not show correlation.
- We model the perturbation strategies according to three different edge modification approaches: Edge add/del, Edge rotation and Edge swap. We also demonstrate that perturbation strategy affects the correlation value between generic and clustering-specific information loss measures.
- Last but not least, we analyse different datasets and prove that correlation depends on dataset properties, as well.

### 1.2 Roadmap

This paper is organized as follows. In Section 2, we review different anonymization processes and some generic measures used for graph assessment. Section 3 presents our experimental framework, including perturbation and clustering methods, graph assessment and data sets used in our experiments. In Section 4, we show the experiments and comment on the results. Finally, in Section 5, we discuss conclusions and future work.

## 2 Anonymization and graph assessment

As we have stated before, the two main objectives of an anonymization process are: (1) to preserve the privacy of users or individuals who appear in a data set, hindering the re-identification processes, and (2) to preserve data utility on anonymized data, i.e., minimizing information loss.

Anonymization methods and graph assessment depend on the type of data they are intended to work with. In this paper, we will work with simple, undirected and unlabelled graphs. Because these graphs have no attributes or labels in the edges, information is only in the structure of the graph itself and, due to this, the adversary can use information about the structure of the network to attack the privacy. However, since all of the information is contained in it, we want to preserve the structure of the graph.

### 2.1 Notation

Let $G = (V, E)$ be a simple graph, where $V$ is the set of nodes and $E$ the set of edges in $G$. We use $v_i \in V$ to denote node $i$ and $(v_i, v_j) \in E$ an edge connecting nodes $v_i$ and $v_j$. We define $n = |V|$ to denote the number of nodes and $m = |E|$ to denote the number of edges. We use $G = (V, E)$ and $\widetilde{G} = (\widetilde{V}, \widetilde{E})$ to indicate the original and the anonymized graphs, respectively.

### 2.2 Anonymization

We categorize anonymization methods on graph formatted data into three main categories:

- Graph modification approaches: These methods anonymize a graph by modifying (adding and/or deleting) edges or nodes in a graph. There are two basic approaches: (1) The simplest way alters the graph structure by removing and adding edges randomly. It is called randomization or random-based approach. (2) Another way consists on edge addition and deletion to fulfil desired constraints, i.e. anonymization methods do not modify edges at random, they modify edges to meet some desired constraints. For example, $k$-anonymity-based approaches modify graph structure (by adding and removing edges) in order to get the $k$-anonymity value for the graph.
- Generalization approaches (also known as clustering-based approaches): These methods cluster nodes and edges into groups. Then, they anonymize each group into a super-node to publish the aggregate information about structural properties of the nodes [16]. The details about individuals can be hidden properly, but the graph may be shrunk considerably after anonymization, which may not be desirable for analyzing local structures.
- Differentially private approaches: These methods refer to algorithms which guarantee that individuals are protected under the definition of differential privacy [11]. Differential privacy imposes a guarantee on the data release mechanism rather than on the data itself. The goal is to provide statistical information about the data while preserving the privacy of users.

Generalization approaches do not enable local structure data analysis, so they are not a good approach to release data for clustering purposes. Differential privacy mechanism provides statistical information about the data, but it does not allow us to release all structural information for clustering purposes. For example, Hay et al. [17] propose an algorithm to public release the degree distribution, which is one of the most commonly studied graph's properties. However, it does not allow us to release the entire network structure. Therefore, we will focus on graph modification approaches, which preserve local structures and keep the details of the data for clustering processes.

One widely adopted strategy of graph modification approaches is randomization. Randomization methods are based on adding random noise in original data. It has been well investigated for relational data. To work with graph data there are two basic approaches [1]: (1) Rand Add/Del: randomly add and delete edges from the original graph (this strategy keeps the number of edges) and (2) Rand Switch: exchange edges between pairs of nodes (this strategy keeps the number of edges and the degree of all nodes). Naturally, edge randomization can also be considered as an additive-noise perturbation.

Hay et al. [15] proposed a method to anonymize unlabelled graphs. This method is called Random Perturbation and is based on removing $p$ edges at random from the graph, and then adding $p$ false edges at random. The set of nodes is not changed and the number of edges is preserved in the anonymized graph.

Ying and Wu [29] studied how different randomization methods (including Rand Add/Del and Rand Switch methods) affect the privacy of the relationship among nodes. After the experiments, they proposed new variations of the randomization algorithms to preserve spectral characteristics of the original graph: Spctr Add/Del and Spctr Switch.

Ying et al. [30] suggested a variation of Rand Add/Del method, called Blockwise Random Add/Delete strategy or simply Rand Add/Del-B. This method divides the graph into blocks according to the degree sequence and implements modifications (by adding and removing edges) on the nodes at high risk of re-identification, not at random over the entire set of nodes. The authors expect to introduce fewer perturbations (with better utility preservation) to achieve the same privacy protection.

Previous methods are all random-based. Another widely adopted strategy of graph modification approaches consists on edge addition and deletion to meet desired constraints. Some desired constraints are based on the $k$-anonymity concept. This concept was introduced by Sweeney [28] for the privacy preservation on relational data. Formally, the $k$-anonymity model is defined as follows: let $RT(A_1,\ldots,A_n)$ be a table and $QI_{RT}$ be the quasi-identifier associated with it. $RT$ is said to satisfy $k$-anonymity if and only if each sequence of values in $RT\left[QI_{RT}\right]$ appears with at least $k$ occurrences in $RT\left[QI_{RT}\right]$. The $k$-anonymity model indicates that an attacker cannot distinguish among different $k$ records although he manages to find a group of quasi-identifiers. Consequently, the attacker cannot re-identify an individual with a probability greater than $\frac{1}{k}$.

The $k$-anonymity model can be applied using different concepts when dealing with networks rather than relational data. A greatly used option is to consider the node degree as a quasi-identifier. This corresponds to $k$-degree anonymity. In short, in $k$-degree anonymity we presume that the only possible attack is when the attacker knows the degree of some nodes. Therefore, if some node is re-identified with this information, then we have an information leakage. $k$-Anonymity methods are based on modifying the network structure (by adding and removing edges) to ensure that all nodes satisfy this model. In other words, the main objective is that all nodes have at least $k-1$ other nodes sharing the same degree. Liu and Terzi [21] developed a method which given a network $G = (V, E)$ and an integer $k$, finds a $k$-degree anonymous network $\widetilde{G} = (V, \widetilde{E})$ where $\widetilde{E} \cap E \approx E$, trying to minimize the number of changes on edges.

Zhou and Pei [34] used the 1-neighbourhood sub-graph of the objective nodes as quasi-identifiers. Let $k$ be a positive integer. For a node $u \in V$, $u$ is $k$-anonymous in anonymization $\widetilde{G}$ if there are at least $k-1$ other nodes $v_1, \ldots, v_{k-1} \in V \mid \Gamma_{\widetilde{G}}(u), \Gamma_{\widetilde{G}}(v_1), \ldots, \Gamma_{\widetilde{G}}(v_{k-1})$ are isomorphic, where $\Gamma(v_i)$ is the 1-neighbourhood of node $v_i$. $G$ is $k$-anonymous if every node in $G$ is $k$-anonymous. It is called $k$-neighbourhood anonymity. Zou et al. [35] considered all structural information about a target node as quasi-identifier and proposed a new model called $k$-automorphism to anonymize a network and ensure privacy against this attack. They define a $k$-automorphic network as follows: given a network $G$, (a) if there exist $k-1$ automorphic functions $F_a(a = 1, \ldots, k-1)$ in $G$, and (b) for each node $v$ in $G$, $F_{a_1}(v) \neq F_{a_2}(1 \leq a_1 \neq a_2 \leq k-1)$, then $G$ is called a $k$-automorphic graph. Hay et al. [16] went a step further. They designed a method, named $k$-candidate anonymity, which uses queries as quasi-identifier. In this method, a node $v_i$ is $k$-candidate anonymous to question $Q$ if there are at least $k-1$ other nodes in the graph with the same answer. Officially, $|cand_Q(v_i)| \geq k$ where $cand_Q(v_i) = \big\{v_j \in V \mid Q(v_j) = Q(v_i)\big\}$. A graph is $k$-candidate anonymous to question $Q$ if all of its nodes are $k$-candidate anonymous to question $Q$. The question $Q$ is modelled according to the knowledge of the adversary assumed.

When there is little diversity in the sensitive attributes inside an equivalence class, it is possible to obtain information from anonymized data. Although there are $k$ indistinguishable records in each equivalence class, if the information in sensitive attributes is the same, it is possible to infer information unless the attacker does not know exactly which record it is. $l$-diversity [23] alleviates the problem of sensitive attribute disclosure. It ensures that the sensitive attribute values in each equivalence class is diverse. An attacker, though, can also infer some sensitive information from similarity or skewness attack [20]. This leads to $t$-closeness [20], which is another privacy definition that considers the sensitive attribute distribution in each class. There are other privacy definitions of this flavour, but they all have been criticized for being ad hoc [33].

### 2.3 Graph assessment

Several generic measures have been used to quantify the structure's properties in graph formatted data. The authors usually use these measures and compare the values obtained by the original and the anonymized data in order to quantify the noise introduced by the anonymization process. When we quantify the information loss as described above, we talk about generic information loss measure.

Hay et al. [15] utilized five structural properties from graph theory for quantifying network structure. For each node, the authors evaluate closeness centrality (average shortest path from one node to every other node), betweenness centrality (proportion of all shortest paths which go through the node) and path length distribution (computed from the shortest path between each pair of nodes). For the graph as a whole, they evaluate the degree distribution and the diameter (the maximum shortest path between two nodes). The objective is to keep these five measures close to their original values, assuming that it involves little distortion in the anonymized data.

Ying and Wu [29] and Ying et al. [30] used both real space and spectrum based characteristics to study how the graph is affected by randomization methods. The authors focused on four real space characteristics of the graph and on two important eigenvalues of the graph spectrum. The real space characteristics are: the harmonic mean of the shortest distance, the modularity (which indicates the goodness of the community structure), the transitivity (which measures the fraction of all possible triangles present in the graph), and the sub-graph centrality (which is used to quantify the centrality of node). Since graph spectrum has close relations with many graph characteristics and can provide global measures for some network properties, the authors also consider the following two spectral characteristics: the largest eigenvalue of the adjacency matrix and the second smallest eigenvalue of the Laplacian matrix.

Alternatively, Zou et al. [35] defined a simple method for evaluating information loss on undirected and unlabelled graphs. The method is based on the difference between the original and the anonymized graph edges, $Cost(G,\widetilde{G}) = (E \cup \widetilde{E}) - (E \cap \widetilde{E})$. Liu and Terzi [21] used clustering coefficient and average path length for the same purpose. Clustering coefficient is the fraction of possible triangles that exist. Average path length is defined as the average number of steps along the all shortest paths.

Hay et al. [16] examined five properties commonly measured and reported on network data: degree (distribution of the degrees of all nodes in the graph), path length (distribution of the lengths of the shortest paths between randomly sampled pairs of nodes), clustering coefficient, network resilience (the number of nodes in the largest connected component of the graph when nodes are removed in degree decreasing order) and infectiousness (measured by calculating the proportion of nodes infected by a hypothetical disease, which is simulated by first infecting a randomly chosen node and then transmitting the disease to each neighbour with the specified infection rate).

There are existing studies that work on graphs trying to maximize some specific task-oriented utility. Budi et al. [3] defined the *kb*-anonymity model, which combines privacy-preserving using the *k*-anonymity model and specific task of behaviour-preserving test and debugging data. Lucia et al. [22] improved the model to avoid the probing attack for evolving programs. Both papers consider the anonymization of paths in a program code, which can be represented as a graph, and use a specific utility measurement, which is test coverage. The aim is to ensure that the replaced data exhibits the same kind of program behaviour shown by the original data so that the replaced data may still be useful for the purposes of testing and debugging. Although there are some similarities between their work and ours, the purpose of ours is quite different.

It is important to emphasize that these generic information loss measures only evaluate structural and spectral changes between original and anonymized data. That is, these measures do not evaluate the data mining processes on anonymized data, and as such, they are general or application-independent. The analysis of specific and application-dependent quality measures is an open problem. We consider in this paper the case of an application in clustering.

## 3 Experimental set up

Some authors evaluate their anonymization methods comparing the results of generic measures on original data with the results on anonymized data. They assume that small distortion on these measures involves little distortion on anonymized data utility. Our objective is to evaluate the correlation between generic information loss (GIL) measures and specific information loss (SIL) measures based on clustering processes. Hence, if the GIL indicates that there is little perturbation on anonymized data, then the clustering results on the anonymized data must be close to the results on the original data. Otherwise, the GIL measures used in graph assessment are not representative of real data utility.

To conduct this experiment, we have to test as much anonymization methods as we can. As we have seen, there are several anonymization methods and it is hard to analyse all of them. Nevertheless, all of graph modification approaches are based on edge modification and can be modelled as an additive-noise perturbation. So, we can model the generic behaviour of these methods through basic edge perturbation. We define the basic edge perturbation on Section 3.2.

Our experimental framework is shown in Figure 1. As we can see, we choose five graph formatted datasets, three edge perturbation methods, several generic information loss measures and six graph clustering algorithms. First, we apply perturbation to graph datasets (details are shown in Section 3.1) using edge perturbation methods (Section 3.2). Then, we evaluate original and perturbed data using GIL measures for quantifying network structure (Section 3.3). Next, we apply the clustering processes (Section 3.4) both on original and on perturbed data and we use clustering-based specific measures (Section 3.5) to evaluate the results. Lastly, we compare the GIL and SIL results. If the degree of similarity between them are close, the GIL measures provide correct information about data utility. Otherwise, these measures do not provide correct information about the utility of the anonymized data for clustering.

Each dataset is perturbed from 1% to 25% of edge set. We compute perturbation percentage using the edge difference (ED), which is defined as the percentage of original edges that are not present in the perturbed graph, as shown in Equation 1.

$$ED(G,\widetilde{G}) = 1 - \frac{|E \cap \widetilde{E}|}{max(|E|,|\widetilde{E}|)} \tag{1}$$

### 3.1 Datasets

Five different real data sets are used in our experiments. Although all these sets are unlabelled, we have selected these datasets because they have different graph properties. They are the following ones:

- **Zachary's karate club** [31] is a small social graph widely used in clustering and community detection. It shows the relationship among 34 members of a karate club.
- **American college football** [12] is a graph of American football games among Division IA colleges during regular season Fall 2000.

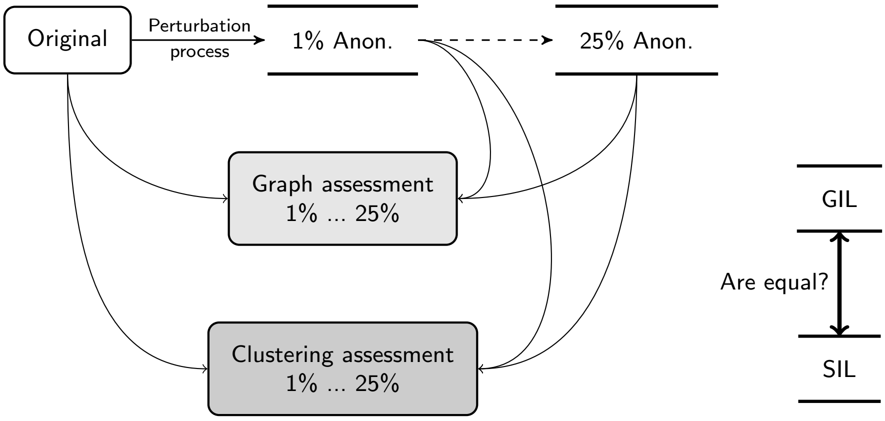
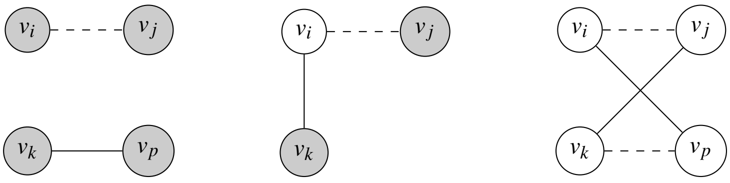


Fig. 1: Experimental framework. Each dataset is perturbed from 1% to 25% using each perturbation method. Next, we compare the original and perturbed data using GIL measures in order to quantify the noise introduced on data. Then, we do the same with real clustering processes and SIL measures. Finally, we compare the results of GIL and SIL measures and evaluate the correlation between them. We want to analyse whether GIL measures are useful to predict the clustering real data utility.

- **Jazz musicians** [13] is a collaboration graph of jazz musicians and their relationship.
- **Flickr** is a sub-graph collected from Flickr OSN. This data has been obtained from [18], where a sampling process has been performed over original data provided by [24]. Nodes represent the users and edges the relationship among them. Although relations are directional in this network, we have eliminated the direction of the edges to get an undirected graph.
- **URV Email** [14] is the email communication network at the University Rovira i Virgili in Tarragona (Spain). Nodes are users and each directed edge represents that at least one email has been sent.

| **Dataset** | $n$ | $m$ | $\overline{deg}$ | $AD$ | $D$ |
|---|---|---|---|---|---|
| Zachary's karate club | 34 | 78 | 4.588 | 2.408 | 5 |
| American college football | 115 | 613 | 10.661 | 2.508 | 4 |
| Jazz musicians | 198 | 2,742 | 27.697 | 2.235 | 6 |
| Flickr | 954 | 9,742 | 20.423 | 2.776 | 4 |
| URV Email | 1,133 | 5,451 | 9.622 | 3.606 | 8 |

Table 1: Datasets' properties. For each dataset we present the number of nodes ($n$), number of edges ($m$), average degree ($\overline{deg}$), average distance (AD) and diameter (D).



(a) Edge add/del (b) Edge rotation (c) Edge swap

Fig. 2: Basic operations for edge modification. Dashed lines represent deleted edges while solid lines are the added ones. Colour of the nodes indicates whether a node changes its degree (grey) or not (white).

Table 1 shows a summary of the datasets' main features, including the number of nodes ($n$), number of edges ($m$), average degree ($\overline{deg}$), average distance (AD) and diameter (D).

### 3.2 Perturbation methods

We model the generic behaviour of the edge-modification methods for graph anonymization through the perturbation introduced by three basic edge modifications. These are:

- **Edge add/del** is the most generic edge modification. It simply consists on deleting an existing edge $(v_i, v_j) \in E$ and adding a new random one $(v_k, v_p) \notin E$. Figure 2a illustrates it.
- **Edge rotation** among three nodes can be defined as follows: if $v_i, v_j, v_k \in V$, $(v_i, v_j) \in E$ and $(v_i, v_k) \notin E$, we delete $(v_i, v_j)$ and create $(v_i, v_k)$. Figure 2b shows this basic operation.
- **Edge swap** between four nodes $v_i, v_j, v_k, v_p \in V$ where $(v_i, v_j)$, $(v_k, v_p) \in E$ and $(v_i, v_p)$, $(v_k, v_j) \notin E$ is defined by deleting edges $(v_i, v_j)$, $(v_k, v_p)$ and creating new edges $(v_i, v_p)$, $(v_k, v_j)$, as we can see in Figure 2c.

As we have already commented, most of these anonymization methods use one (or more) of these basic edge modification or perturbation. It is true that some anonymization methods do not apply edge modification over all edge set, but this behaviour is specific and different for each anonymization method. We believe that this approach can model the basic behaviour of edge-modification methods for graph anonymization, although each method has its specific peculiarities.

For all perturbation methods, the number of nodes and edges remain the same, but the degree distribution changes on Edge add/del and Edge rotation, while it remains the same on Edge swap. Clearly, Edge add/del is the most general concept and all other perturbations can be modelled as a particular case of Edge add/del. Therefore, Edge rotation is a subset of Edge add/del and Edge swap is a subset of Edge rotation, being the most specific concept.

All random-based anonymization methods related to Rand Add/Del are clearly related to Edge add/del perturbation concept. For example, Random Perturbation algorithm [15], Spctr Add/Del [29] and Rand Add/Del-B [30] use this concept to anonymize graphs. Most of *k*-anonymity methods can be also modelled through Edge add/del concept [34, 35, 16]. Edge rotation is a specification of Edge add/del and a generalization of Edge swap. On every edge movement, one node keeps its degree and the other changes it. The Univariant Micro-aggregation for Graph Anonymization algorithm (UMGA) [5] applies this concept to anonymize the graph according to *k*-degree anonymity concept. Other methods are related to Edge swap perturbation concept. For instance, Rand Switch and Spctr Switch [29] apply this concept to anonymize a graph. Liu and Terzi [21] also apply this concept to graph's reconstruction step of their algorithm for *k*-degree anonymity.

### 3.3 Graph assessment

We use different generic measures for quantifying network structure. These generic measures are used to compare both the original and the anonymized data to quantify the noise introduced in the perturbed data by the anonymization process. These generic measures evaluate some key graph's properties. They evaluate the graph structure, so they are general or, in other words, application-independent. Information loss was defined by the discrepancy between the results obtained on the original and the anonymized data.

In our experiments we use several graph measures based on structural and spectral properties. In the rest of this section we review the measures used.

**Average distance** (AD) is defined as the average of the distances between each pair of nodes in the graph. It measures the minimum average number of edges between any pair of nodes. Formally, it is defined as:

$$AD(G) = \frac{\sum_{i,j} d_{ij}}{\binom{n}{2}} \tag{2}$$

where $d_{ij}$ is the length of the shortest geodesic path from $v_i$ to $v_j$, meaning the number of edges along the path.

**Diameter** [15] (D) is defined as the largest minimum distance between two nodes in the graph, as Equation 3 shows.

$$D(G) = max(d_{ij}), \forall i \neq j \tag{3}$$

Another used measure is **edge intersection** [35, 21] (EI). It is defined as the percentage of original edges which are also in the anonymized graph. Formally:

$$EI(G, \widetilde{G}) = \frac{|E \cap \widetilde{E}|}{max(|E|, |\widetilde{E}|)} \tag{4}$$

**Clustering coefficient** [21, 16, 12, 6] (C) is a measure widely used in the literature. The clustering coefficient of a graph is the average:

$$C(G) = \frac{1}{n} \sum_{i=1}^{n} C(v_i) \tag{5}$$

where $C(v_i)$ is the clustering coefficient for node $v_i$. The clustering of each node is the fraction of possible triangles that exist. For each node the clustering coefficient is defined by:

$$C(v_i) = \frac{2T(v_i)}{deg(v_i)(deg(v_i)-1)} \tag{6}$$

where $T(v_i)$ is the number of triangles surrounding node $v_i$, and $deg(v_i)$ is the degree of $v_i$.

**Transitivity** [29, 30, 6] (T) is the fraction of all possible triangles present in the graph. Possible triangles are identified by the number of triads (two edges with a shared node), as we can see in Equation 7.

$$T(G) = \frac{3 \times (number of triangles)}{(number of triads)} \tag{7}$$

**Betweenness centrality** [15] (BC) is a centrality measure, which calculates the fraction of number of shortest paths that go through each node. This measure indicates the centrality of a node based on the flow among other nodes in the graph. A node with a high value indicates that this node is part of many shortest paths in the graph, which will be a key node in the graph structure. We define the betweenness centrality of a node $v_i$ as:

$$BC(v_i) = \frac{1}{n^2} \sum_{s,t} \frac{g_{st}^i}{g_{st}} \tag{8}$$

where $g_{st}^i$ is the number of geodesic paths from $v_s$ to $v_t$ that pass through $v_i$, and $g_{st}$ is the total number of geodesic paths from $v_s$ to $v_t$.

The second centrality measure is **closeness centrality** [15] (CC), which is described as the inverse of the average distance to all accessible nodes. Closeness is an inverse measure of centrality in which a larger value indicates a less central node, while a smaller value indicates a more central node. Formally, we define the closeness centrality of a node $v_i$ as:

$$CC(v_i) = \frac{n}{\sum_j d_{ij}} \tag{9}$$

And the last centrality measure is **degree centrality** [15] (DC). It evaluates the centrality of each node associated with its degree. That is, the fraction of nodes connected to it. A higher value indicates greater centrality in the graph. The degree centrality of a node $v_i$ is depicted in Equation 10.

$$DC(v_i) = \frac{deg(v_i)}{m} \tag{10}$$

The last three centrality measures described above evaluate the centrality of each node of the graph from different concepts. These measures give us a value of centrality for each node. To assess the perturbation introduced in the graph by the anonymization process, we compute the vector of differences for each node between the original and the anonymized graph. Then, we compute the root mean square (*RMS*) to obtain a single value for the whole graph. We calculate the difference of the centrality measures between the original and the anonymized graph as follows:

$$\varepsilon(G,\widetilde{G}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(g_i - \widetilde{g}_i)^2} \tag{11}$$

where $g_i$ is the value of the centrality measure for the node $v_i$ of $G$, and $\widetilde{g}_i$ is the value of the centrality measure for the node $v_i$ of $\widetilde{G}$. In our experiments we use Equation 11 to compute a value representing the error induced in the whole graph by the anonymization process in the centrality measures.

We also focus on two important eigenvalues of the graph spectrum. The first one is the **largest eigenvalue of the adjacency matrix A** ($\lambda_1$) [29] where $\lambda_i$ are the eigenvalues of A and $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. The eigenvalues of A encode information about the cycles of a graph as well as its diameter. The spectral decomposition of A is:

$$A = \sum_i \lambda_i e_i e_i^T \tag{12}$$

where $e_i$ is the eigenvector corresponding to $\lambda_i$ eigenvalue.

The other one is the **second smallest eigenvalue of the Laplacian matrix L** ($\mu_2$) [29], where $\mu_i$ are the eigenvalues of L and $0 = \mu_1 \leq \mu_2 \leq \ldots \leq \mu_m \leq m$. The eigenvalues of L encode information about the tree structure of G. $\mu_2$ is an important eigenvalue of the Laplacian matrix and can be used to show how good the communities separate, with smaller values corresponding to better community structures. Laplacian matrix is defined as:

$$L = D - A \tag{13}$$

where $D_{n\times n}$ is a diagonal matrix with row-sums of A along the diagonal, and 0's elsewhere.

The number of nodes, edges and average degree are not considered as parameters to assess anonymization process, since anonymization methods analysed in this work keep these values constant.

### 3.4 Clustering methods

Six clustering algorithms are used to evaluate the perturbation methods. All of them are unsupervised algorithms for graph formatted data based on different concepts and developed for different applications and scopes. An extended revision and comparison of them, among others, can be found at [19, 32]. The selected clustering algorithms are:

- Markov Cluster Algorithm (**MCL**) was developed by S. Van Dongen [10]. The algorithm is based on the simulation of flow in graphs and it is widely used in bioinformatics. It starts by computing an integer power of the diffusion matrix (usually the square), which yields the probability matrix of a random walk after a specific number of steps. This step is called expansion. Next, it computes the probability of the walker to be trapped within a community. This step is called inflation. The expansion and inflation steps are iterated until it obtains a disconnected tree. Its components are the communities. The *inflation* parameter controls the granularity of the result sets, and its value is adjusted according to the data of each graph. Its complexity can be lowered to $\mathcal{O}(Nk^2)$ if, after each inflation steps, only the k largest elements of the resulting matrix are kept, whereas the others are set to zero.

- Algorithm of Girvan and Newman (**Girvan-Newman** or **GN**) [25] is an important community detection algorithm in graphs. It is a hierarchical divisive algorithm, in which edges are iteratively removed based on the value of their betweenness centrality. The algorithm has a complexity $\mathcal{O}(N^3)$ on a sparse graph.
- Fast greedy modularity optimization (**Fastgreedy**) by Clauset, Newman and Moore [7] is a hierarchical agglomeration algorithm for detecting community structure. Starting from a set of isolated nodes, the edges of the original graph are iteratively added to produce the largest possible increase of the modularity at each step. Its running time on a sparse graph is $\mathcal{O}(N \log^2 N)$.
- **Walktrap** [26] by Pons and Latapy tries to find densely connected sub-graphs, also called communities in a graph via random walks. The idea is that short random walks tend to stay in the same community. They proposed a measure of similarities between nodes based on random walks to capture the community structure in a graph. It runs in time $\mathcal{O}(mn^2)$ and space $\mathcal{O}(n^2)$ in the worst case.
- **Infomap** by Rosvall and Bergstrom [27] use the problem of optimally compressing the information on the structure of the graph to find the best cluster structure. This is achieved by compressing the information of a dynamic process taking place on the graph, namely a random walk. The optimal compression is achieved by optimizing a quality function, which is the Minimum Description Length of the random walk. Such optimization can be carried out rather quickly with a combination of greedy search and simulated annealing.
- **Multilevel** by Blondel et al. [2] is a multi-step technique based on a local optimization of Newman-Girvan modularity in the neighbourhood of each node. After a partition is identified in this way, communities are replaced by super-nodes, yielding a smaller weighted network. The procedure is then iterated, until modularity does not increase any further. The computational complexity is essentially linear in the number of edges of the graph.

MCL, Walktrap and Infomap are based on the random walk concept, while Girvan-Newman and Fastgreedy are based on hierarchical edge betweenness, and Multilevel is based on modularity concept. Although some algorithms permit overlapping among different clusters, we have not allowed such overlapping in our experiments by setting the parameter to zero. This is because setting overlapping to zero simplifies the method which evaluates the similarity between results.

### 3.5 Clustering assessment

In this work we want to analyse the utility of the perturbed data by evaluating it on different clustering processes. Like generic graph measures, we compare the results obtained both by the original and the perturbed data in order to quantify the level of noise introduced in the perturbed data. This measure is specific and application-dependent, but it is necessary to test the perturbed data in real clustering processes.

We consider the following approach to measure the clustering assessment for a particular perturbation and clustering method: (1) apply the perturbation method $p$ to the original data $G$ and obtain $\widetilde{G}$; (2) apply a particular clustering method $c$ to $G$ and obtain clusters $c(G)$ and apply the same method to $\widetilde{G}$ to obtain $c(\widetilde{G})$; (3) compare the clusters $c(G)$ to $c(\widetilde{G})$. In relation to information loss, it is clear that the more similar $c(\widetilde{G})$ is to $c(G)$, the less information loss. Thus, clustering specific information loss measures should evaluate the divergence between both sets of clusters $c(G)$ and $c(\widetilde{G})$.

Ideally, results should be the same. That is, the same number of sets with the same elements in each set. In this case, we can say that the anonymization process has not affected the clustering process. When the sets do not match, we should be able to calculate a measure of divergence.

For this purpose, we use **precision** index [4]. Under the situation that true communities of a graph are known a priori, precision index could be directly used to evaluate the similitude between two sets of clusters. Given a graph composed of $n$ nodes and $m$ communities, each community is assigned label $l_{tc}$. Nodes are assigned the same labels $l_{tc}$ as the community they belong to, where $l_{tc}$ is the true label for each node. In our experiments, the true communities are the ones assigned by the original dataset. Assuming the graph has been divided into clusters, for every cluster $i$, we examine all nodes in $i$ and obtain the frequency that the true labels occur. The label that most frequently occurs is assigned as the predicted label $l_{pc}$ to each node in the cluster $i$. The precision is then defined as the fraction of all nodes in which the predicted label $l_{pc}$ is the same as the true label $l_{tc}$:

$$Precision = \frac{\sum_{v=1}^{n} equal(l_{tc}, l_{pc})}{n} \tag{14}$$

where $equal(x,y) = 1$ if $x = y$ and 0 otherwise.

Notice that the precision is a value in the range [0,1], which takes the value 0 when there is no overlap between the sets and the value 1 when the overlap between the sets is complete.

## 4 Experimental results

To compare the cluster-specific measures and the generic ones, we have computed these measures for pairs of graphs ($G$, $\widetilde{G}$) using some particular perturbation method $p$. That is $\widetilde{G} = p(G)$.

In this section, we show the results of our experiments. For each dataset ($G$) we apply the three perturbation methods, and then, we assess graph measures on the perturbed data. Next, we apply clustering algorithms and compare the original and the perturbed results using the precision index, our cluster-based measure. We refer to specific information loss measure as a result of precision index applied to specific clustering algorithm on original and anonymized data. In other words, we refer to specific information loss measure as a value of $Precision(c(G), c(\widetilde{G}))$, where $c$ is one of our clustering methods and $\widetilde{G} = p(G)$ for a particular perturbation method $p$. From now on, we will use the name of the clustering algorithm to refer to its precision index computed as we have mentioned above.

Perturbation methods have been applied with a percentage of noise. It has been added iteratively with a 1% of the number of edges at each step. The goal is that we can see how the structural properties of the graph evolve. The values showed by all measures and metrics are averaged values computed from independent executions. In our framework, we have used the Pearson correlation to compute the linear dependence between all measures and metrics, where the $p$-values refer to the observed significance level of a hypothesis test with the *null hypothesis* that correlation is equal to 0.

Parameters used in our experiments are detailed in Table 2. “Perturbation range” specifies the percentage of noise introduced, “execs” is the number of independent executions for every experiment and “MCL inflation” parameter controls the granularity of the resulting clusters on Markov Cluster Algorithm (MCL). Others clustering methods are used with default values.

| **Parameter** | **Value** |
|---|---|
| Perturbation range | From 1% to 25% |
| Execs | 20 |
| MCL Inflation | 1.8 |

Table 2: Parameters used in our experiments.

In our experiments we want to address the following questions:

- Do the generic information loss measures and precision behave in similar way independently of the dataset? In Section 4.1 we analyse whether the GIL and SIL measures present similar behaviour over different datasets, i.e., they behave in similar way independently of the specific characteristics of the dataset.
- Are the generic information loss measures correlated with clustering-specific measures? In Section 4.2 we compare the GIL versus SIL measures in order to describe the correlation among them.
- What are the effects of various data perturbation strategies? In Section 4.3 we comment the correlation’s results based on the three perturbation methods presented on Section 3.2.
- What are the differences between measures when various graph datasets are considered? In Section 4.4 we discuss the correlation’s results for each specific dataset to analyse the differences among them.

### 4.1 Analysing measures

Before comparing the generic information loss measures with the precision, we analyse all measures in order to evaluate whether each measure behaves in similar way to itself on different datasets. If a measure presents a high self-correlation, then it will conduct in a similar way independently of the data where it is applied. Therefore, we analyse the self-correlation of all generic and clustering-specific measures. Self-correlation is measured by comparing the results of specific measure over each dataset to all others. Then, we calculated the Pearson correlation on the resulting set.

| **Pearson** | *AD* | *D* | *BC* | *CC* | *DC* | *EI* | *C* | *T* | $\lambda_1$ | $\mu_2$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $r$ | 0.846 | 0.150 | 0.957 | 0.902 | 0.992 | 0.999 | 0.974 | 0.948 | 0.247 | 0.097 |
| $p$-value | *(0.000)* | *(0.007)* | *(0.000)* | *(0.000)* | *(0.000)* | *(0.000)* | *(0.000)* | *(0.000)* | *(0.000)* | *(0.006)* |

Table 3: Pearson self-correlation value ($r$) and its observed significance level ($p$-value) of generic information loss (GIL) measures.

Most of the GIL measures show strong self-correlation over all datasets used in our experiments. As we can see in Table 3, average distance (AD), betweenness centrality (BC), closeness centrality (CC), degree centrality (DC), edge intersection (EI), clustering (C), and transitivity (T) present self-correlation values higher than 0.84 with $p$-values equal to 0. These results confirm that the measures evolve in a similar way over all datasets, i.e., the behaviour of the measures is similar independently of the dataset in which they are applied. Diameter, the largest eigenvalue of the adjacency matrix ($\lambda_1$) and the second smallest eigenvalue of the Laplacian matrix ($\mu_2$) present weak and very weak self-correlation values. It denotes that their behaviour is clearly subordinate to the dataset. It is interesting to underline that diameter cannot be computed on Flickr and URV Email datasets because the perturbation methods generate some isolated nodes, so the number of connected components is greater than one and the diameter of the dataset cannot be computed. Furthermore, on Football dataset the diameter keeps the same value on all perturbed data. That is because Football dataset does not follow the power-law on degree distribution. All nodes have a degree value between 7 and 12, and it is improbable to increase or decrease the diameter value by random edge perturbation. Therefore, the behaviour of the diameter on a perturbation process is very dependent of the dataset.

| **Pearson** | *MCL* | *Infomap* | *Multilevel* | *GN* | *Fastgreedy* | *Walktrap* |
|---|---|---|---|---|---|---|
| $r$ | 0.287 | 0.626 | 0.777 | 0.828 | 0.782 | 0.656 |
| $p$-value | *(0.000)* | *(0.000)* | *(0.000)* | *(0.000)* | *(0.000)* | *(0.000)* |

Table 4: Pearson self-correlation value ($r$) and its observed significance level ($p$-value) of precision index, our clustering-specific information loss measure. We use the name of the clustering algorithm to refer to its precision value computed as we have described in Section 4.1.

The precision values are presented in Table 4. As we have mentioned, we use the name of the clustering algorithm to refer to its specific information loss measure computed as $Precision(c(G), c(\widetilde{G}))$, where $c$ is one of our clustering methods and $\widetilde{G} = p(G)$

for a particular perturbation method $p$. Most of them show strong self-correlation values and all of them present $p$-values equal to 0. Multilevel, Girvan-Newman and Fastgreedy achieve self-correlation values greater than 0.77, while Infomap and Walktrap achieve values close to 0.6, and MCL presents weak self-correlation with a value of 0.28.

### 4.2 Comparing generic and clustering-specific measures

Pearson correlation values for all generic information loss and clustering-specific measures are shown in Table 5. These values are computed using all datasets and all perturbation methods.

The first and the second generic information loss measures which we analyse are average distance and diameter. Both measures are related to paths and cannot be calculated for graphs with two or more connected components. Thus, both measures cannot be computed on few perturbed graphs because isolated nodes have appeared during perturbation process. The Pearson correlation index for average distance and the precision for all clustering methods is 0.732. As we can see in Table 5, average distance achieves strong correlation values with all clustering methods, except with MCL, where the achieved value is quite lower than others and indicates a moderate correlation between average distance and the precision values of MCL clustering. As we can see, $p$-values are 0 for all experiments, demonstrating that results are statistically significant. On the contrary, diameter does not present correlation since its $p$-values are greater than 0.05 on three experiments, and no statistical significance can be assigned to its correlation value. In addition, the correlation value is 0.128, which is very low.

Two of the three centrality measures show similar behaviour and achieve strong correlation values. The correlation index is 0.753 for betweenness centrality, 0.848 for closeness centrality and 0.422 for degree centrality. Therefore, betweenness and closeness centrality are strong correlated to clustering-based measures. Betweenness and closeness centrality achieve correlation values higher than 0.831 for Multilevel, Girvan-Newman and Fastgreedy clustering algorithm. Clearly, the Girvan-Newman algorithm are related to betweenness centrality since it uses the edge betweenness values to discover the clusters, and the edge betweenness and the node betweenness are related. Multilevel and Fastgreedy use the concept of modularity, which is also related to node centrality. The other centrality measure, degree centrality, presents different behaviour and a moderate correlation value of 0.422. Our perturbation methods introduce different noise on degree centrality measure: Edge add/del modify the degree of four nodes, Edge rotation modify the degree of only two nodes and Edge swap does not modify the degree of any node. Thus, Edge swap keeps the same values for this measure but introduce noise on anonymized data and perturb the precision values with all clustering methods. Similar situation is presented on Edge rotation. Clearly, degree centrality is not a suitable measure to analyse the noise introduced by perturbation methods using both Edge rotation or Edge swap.

Edge intersection is a simple measure which is strong correlated to clustering-based measures. Clustering coefficient and transitivity are also strong correlated to clustering-based measures. These measures show strong correlation to the precision results of Infomap, Multilevel, Girvan-Newman, Fastgreedy and Walktrap. As the measures we have seen in the previous paragraph, these ones show low correlation to the precision results of MCL. The mean correlation indexes are 0.785, 0.814 and 0.743 for edge intersection, clustering coefficient and transitivity.

Next, we will analyse the spectrum-based generic information loss measures. These are the largest eigenvalue of the adjacency matrix and the second smallest eigenvalue of the Laplacian matrix. The largest eigenvalue of the adjacency matrix presents moderate correlation, achieving a value of 0.442. The second eigenvalue of the Laplacian matrix does not get good results on any clustering-based measures, achieving an averaged value of 0.116. Additionally, the $p$-values demonstrate the results are not statistically significant.

| **Pearson** | *MCL* | *Infomap* | *Multilevel* | *GN* | *Fastgreedy* | *Walktrap* | $\mu$ |
|---|---|---|---|---|---|---|---|
| AD | 0.580<br>*(0.000)* | 0.716<br>*(0.000)* | 0.807<br>*(0.000)* | 0.785<br>*(0.000)* | 0.747<br>*(0.000)* | 0.755<br>*(0.000)* | 0.732 |
| D | 0.201<br>*(0.000)* | 0.101<br>*(0.075)* | 0.098<br>*(0.083)* | 0.134<br>*(0.018)* | 0.218<br>*(0.000)* | 0.014<br>*(0.803)* | 0.128 |
| BC | 0.559<br>*(0.000)* | 0.687<br>*(0.000)* | 0.854<br>*(0.000)* | 0.865<br>*(0.000)* | 0.831<br>*(0.000)* | 0.724<br>*(0.000)* | 0.753 |
| CC | 0.667<br>*(0.000)* | 0.833<br>*(0.000)* | 0.903<br>*(0.000)* | 0.909<br>*(0.000)* | 0.874<br>*(0.000)* | 0.899<br>*(0.000)* | 0.848 |
| DC | 0.296<br>*(0.000)* | 0.380<br>*(0.000)* | 0.416<br>*(0.000)* | 0.504<br>*(0.000)* | 0.481<br>*(0.000)* | 0.457<br>*(0.000)* | 0.422 |
| EI | 0.581<br>*(0.000)* | 0.820<br>*(0.000)* | 0.861<br>*(0.000)* | 0.887<br>*(0.000)* | 0.814<br>*(0.000)* | 0.748<br>*(0.000)* | 0.785 |
| C | 0.614<br>*(0.000)* | 0.833<br>*(0.000)* | 0.889<br>*(0.000)* | 0.909<br>*(0.000)* | 0.836<br>*(0.000)* | 0.802<br>*(0.000)* | 0.814 |
| T | 0.557<br>*(0.000)* | 0.763<br>*(0.000)* | 0.840<br>*(0.000)* | 0.840<br>*(0.000)* | 0.770<br>*(0.000)* | 0.690<br>*(0.000)* | 0.743 |
| $\lambda_1$ | 0.191<br>*(0.000)* | 0.482<br>*(0.000)* | 0.509<br>*(0.000)* | 0.546<br>*(0.000)* | 0.529<br>*(0.000)* | 0.397<br>*(0.000)* | 0.442 |
| $\mu_2$ | 0.086<br>*(0.088)* | 0.152<br>*(0.003)* | 0.131<br>*(0.010)* | 0.154<br>*(0.002)* | 0.135<br>*(0.007)* | 0.040<br>*(0.429)* | 0.116 |
| $\mu$ | 0.433 | 0.577 | 0.631 | 0.653 | 0.624 | 0.553 | NA |

Table 5: The Pearson correlation values ($r$) between clustering precision value and generic information loss measures (average distance (AD), diameter (D), betweenness centrality (BC), closeness centrality (CC), degree centrality (DC), edge intersection (EI), clustering coefficient (C), transitivity (T), the largest eigenvalue of the adjacency matrix ($\lambda_1$) and the second smallest eigenvalue of the Laplacian matrix ($\mu_2$)). The $p$-values for each correlation are showed within brackets and the last column and row show the average values for each row and column ($\mu$).

Finally, it is important to underline that the precision on clustering measures achieves moderate correlation, with values from 0.624 to 0.653, on Multilevel, Girvan-Newman and Fastgreedy algorithms. Infomap and Walktrap achieve lower correlation values, but still moderate correlation. MCL presents the worst results of all clustering algorithms with a correlation value of 0.433.

**Aggregating some generic measures** In the previous paragraphs we have compared the correlation between each individual generic information loss measures and the precision index. Here, we consider an overall assessment between a group of some GIL measures and the precision. This experiment tries to illustrate which group of one or more GIL measures are the best ones to explain the clustering-based measures.

| **Num.** | *GIL measures* | *r-square* | $\sigma$ |
|---|---|---|---|
| 1 | CC | 0.725 | 0.146 |
| 2 | BC+CC | 0.742 | 0.150 |
| 3 | BC+CC+EI | 0.765 | 0.155 |
| 4 | D+BC+CC+EI | 0.777 | 0.127 |
| 5 | AD+D+BC+CC+EI | 0.787 | 0.117 |

Table 6: Results of regression analysis, where the dependent variable is the precision index and the independent variable is a set of one or more GIL measures. The first column indicates the number of GIL measures considered, the second one the GIL measures set which achieves the best result, the third column the $r$-square value, and the last one the standard deviation ($\sigma$).

Let us consider the $r$-square from a multivariate regression analysis, where the dependent variable is the precision index and the independent variable is a set of one or more GIL measures. The $r$-square value is indicative of the aggregate correlation between a set of the GIL measures and the precision values. We compute the regression analysis between all combinations from 1 to 5 GIL measures and the precision. The result for only one measure is perfectly consistent with Pearson correlation analysis, which has explained previously, where closeness centrality achieves the best result, as we can see in Table 6. The best combination of two GIL measures is betweenness and closeness centrality, with a $r$-square value of 0.742 and standard deviation of 0.15. For three GIL measures, the best combination is the betweenness centrality, closeness centrality and edge intersection. All GIL measures have obtained high individual correlation values, therefore these results are predictable. Nevertheless, when we consider the best combination of four GIL measures, the diameter appears, which has obtained very low individual correlation values. It is interesting because diameter can help us to predict the clustering-specific perturbation in combination with other GIL measures, but it can be useless when we consider only the diameter. Finally, average distance is added to the group when considering a combination of five measures. The $r$-square value is 0.787. It is relevant to note that if more measures are considered, higher $r$-square values are obtained. Thus, the $r$-square difference between 2 and 5 measures is close to 0.04, which implies a little gain and also a considerable increment of computational and time cost. The problem and its peculiarities are the key points to determine the best combination (one or more) of GIL measures to predict the clustering-specific ones.

### 4.3 Comparing perturbation methods

Next, we will briefly analyse the results based on perturbation methods. Edge add/del is the most general edge modification. It adds and deletes edges at random over entire nodes set. Therefore, an edge is created in every step and another is deleted, changing the degree of four nodes (two nodes decrease their degree while two others increase theirs). Hence, the degree sequence and all related measures suffer high perturbation when Edge add/del is applied. Edge rotation is a sub-set of Edge add/del. It modifies an edge keeping one node and changing the other. Thus, two nodes change their degree in every step (one node decreases its and another increases its). Therefore, the degree sequence and related measures suffer quite less perturbation than Edge add/del. Finally, Edge swap switches two edges between four nodes, but none of them modify their degree. Accordingly, the degree sequence and related measures do not modify their values. It is a sever problem, because these measures do not transmit the noise introduced on data.

The degree centrality measure evaluates the centrality of each node associated with its degree. Edge swap does not change the node's degree, so it does not introduce perturbation on this measure. But the precision for all clustering methods on all datasets show that Edge swap introduces perturbation on data. Therefore, this measure is not correlated with clustering results and it is not a suitable measure to evaluate the perturbation introduced in the perturbed data.

| **Pearson** | *Edge add/del* | *Edge rotation* | *Edge swap* |
|---|---|---|---|
| $\mu$ | 0.670 | 0.698 | 0.705 |
| $\sigma$ | 0.206 | 0.208 | 0.211 |

Table 7: Pearson correlation averaged values ($\mu$) and standard deviation ($\sigma$) for each perturbation method (values are averaged over all generic and clustering-based information loss and all datasets).

Table 7 presents average results and standard deviation for each perturbation method. For each method we compute the correlation between all generic and specific information loss measures over all datasets, and then we calculate the averaged value and the standard deviation for all values which are statistically significant (i.e., for all correlation values with $p$-value $< 0.05$). It is interesting to note that Pearson correlation values obtained by Edge swap are higher, on almost all cases, than Edge rotation and Edge add/del. The average value is 0.705 for Edge swap, while Edge rotation gets a value of 0.698 and Edge add/del a value of 0.670. The standard deviation is similar in all methods.

### 4.4 Comparing datasets

In this section, we will summarise the results based on datasets. Table 8 presents the Pearson correlation averaged values and standard deviation for each dataset used in our experiments. For each dataset we compute the correlation between all generic and clustering-based information loss measures over all perturbation methods, and then we calculate the averaged value and the standard deviation for all values which are statistically significant (i.e., for all correlation values with $p$-value $< 0.05$). We can see important differences between datasets. For example, correlation values between generic information loss measures and precision on American college football (Football) achieve the highest correlation value and the lowest standard deviation value, while on Zachary's karate club (Karate) the value keeps quite low and the standard deviation value rises. As we have commented, Football dataset is a collaboration network representing American football games among colleges during regular season. Hence, it does not follow the power-law on degree sequence. The minimum degree value is 7 and the maximum is 12. Therefore, the connectivity is homogeneous in this graph since there is no hubs and all nodes are highly connected to other nodes. Probably, this graph is more robust to noise than other graphs and the perturbation methods do not cause abruptly disruption on perturbed data.

| **Pearson** | *Karate* | *Football* | *Jazz* | *Flickr* | *URV Email* |
|---|---|---|---|---|---|
| $\mu$ | 0.716 | 0.796 | 0.717 | 0.780 | 0.729 |
| $\sigma$ | 0.247 | 0.119 | 0.170 | 0.184 | 0.163 |

Table 8: Pearson correlation averaged values ($\mu$) and standard deviation ($\sigma$) for each dataset (values are averaged over all generic and clustering-specific information loss and all perturbation methods).

### 4.5 Summary

Firstly, we have analysed the behaviour of the generic information loss measures and precision over different datasets. As we have seen, some measures behave in similar way independently of the data where they are applied. Only diameter, the largest eigenvalue of the adjacency matrix and the second smallest eigenvalue of the Laplacian matrix present weak self-correlation values, indicating that their behaviour is dependent on the data where they are applied.

Secondly, we have compared the generic information loss measures and the precision index. Our experiments are based on correlation between each generic information loss measure and precision index computed for each clustering algorithm. The tests showed strong correlation between some generic information loss measures and the clustering-based ones. Some of those discussed are the average distance, betweenness centrality, closeness centrality, edge intersection, clustering coefficient, and transitivity. Degree centrality and the largest eigenvalue of the adjacency matrix present moderate correlation values. As we have mentioned, these measures are clearly subordinate to the perturbation method applied, and therefore the correlation is lower than other measures. Finally, some generic information loss measures, like diameter and the second smallest eigenvalue of the Laplacian matrix, show weak correlation values with the clustering-specific measures. In addition, some experiments related with these measures are not statistically significant.

Thirdly, we have demonstrated that considering two or more generic information loss measures helps us to get a higher correlation value. Therefore, the gain is not great and the complexity rises when considering two or more generic measures.

Fourthly, we have exposed that the perturbation method affects the correlation between generic and cluster-specific information loss measures. As we have seen, datasets perturbed by Edge swap show higher correlation between generic and specific information loss measures than the datasets perturbed by Edge add/del or Edge rotation.

Finally, we have discussed the effect of the datasets on correlation between generic and clustering-specific information loss measures. We have seen substantial differences between datasets in our experimental framework. The degree distribution and the connectivity of the graph affect the robustness of the network and how the perturbation affects the graph's structure.

## 5 Conclusions and further research

In this paper we have reported an experimental study of the possible correlation between generic information loss measures and the clustering-specific ones. We have applied three perturbation methods based on randomization techniques on five real networks. We have used a graph collected from Flickr OSN, two real world social networks that have well-known documented structures (Zachary's karate club and American college football) and two collaboration networks (Jazz musicians and URV Email). We have studied different perspectives of randomization, from graph assessment to clustering assessment.

After seeing the results of the experiments, we can see that there are strong correlations between some generic information loss measures and precision index, our clustering-specific measure. They are average distance, betweenness centrality, closeness centrality, edge intersection, clustering coefficient and transitivity. Other measures, degree centrality and the largest eigenvalue of the adjacency matrix, present moderate correlation values. However, we have not found clear correlations between the precision index and the diameter and the second smallest eigenvalue of the Laplacian matrix. Some experiments related to these two measures are not statistically significant, and in addition, the correlation values are very low. Considering two or more generic information loss measures is possible, thus the complexity rises and the correlation increments only a little.

Even so, we have seen that datasets and randomization methods are significant to determine the correlation between generic and cluster-specific information loss in some cases. Clearly, the edge modification method and the structure and properties of the dataset are playing an important role on anonymization results and data utility. Nevertheless, it is also true that there is an important correlation between some generic information loss measures and the cluster-specific ones, independently of the dataset and the perturbation method applied.

Certainly, the purpose of the data should be taken into account during the anonymization process. Each dataset has its own properties which should be analysed to choose the best anonymization method. If different datasets are generated according to each problem-specific environment, then it is necessary to analyse the background knowledge an attacker can infer from different anonymized datasets.

Many interesting directions for future research have been uncovered by this work. Other measures of quality should be evaluated. For example, other spectral properties can be considered as graph assessment measures. Another data mining processes should be also used to evaluate anonymized data. Finally, other graph's types will be considered, such as weighted graphs [9], directed graphs or bipartite graphs [8].

**Acknowledgements** This work was partly funded by the Spanish Government through projects TIN2011-27076-C03-02 "CO-PRIVACY", CONSOLIDER INGENIO 2010 CSD2007-0004 "ARES" and TIN2010-15764 "N-KHRONOUS".

## References

1. Aggarwal CC and Wang H (eds) (2010) Managing and Mining Graph Data. Springer, New York
2. Blondel VD, Guillaume J-L, Lambiotte R and Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech 2008(10):P10008
3. Budi A, Lo D, Jiang L and Lucia (2011) *kb*-Anonymity: A model for anonymized behaviour-preserving test and debugging data. In: ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI). ACM Press, New York, pp 447-457
4. Cai B-J, Wang H-Y, Zheng H-R and Wang H (2010) Evaluation repeated random walks in community detection of social networks. In: 2010 International Conference on Machine Learning and Cybernetics (ICMLC). IEEE Computer Society, Qingdao, pp 1849-1854
5. Casas-Roma J, Herrera-Joancomartí J and Torra V (2013) An Algorithm For *k*-Degree Anonymity On Large Networks. In: Proceedings of the 2013 International Conference on Advances on Social Networks Analysis and Mining (ASONAM). IEEE Computer Society, Niagara Falls, pp 671-675
6. Chakrabarti D and Faloutsos C (2006) Graph mining: Laws, generators, and algorithms. ACM Comput Surv 38(1):2:1-2:69
7. Clauset A, Newman MEJ and Moore C (2004) Finding community structure in very large networks. Phys Rev E 70(6):066111
8. Cormode G, Srivastava D, Yu T and Zhang Q (2010) Anonymizing bipartite graph data using safe groupings. Proc VLDB Endow 19(1):115-139
9. Das S, Egecioglu Ö and Abbadi A (2010) Anonymizing weighted social network graphs. In: IEEE 26th International Conference on Data Engineering (ICDE). IEEE Computer Society, Long Beach, pp 904-907
10. Dongen S-M (2000) Graph clustering by flow simulation. Dissertation, University of Utrecht.
11. Dwork C (2006) Differential Privacy. In: Proceedings of the 33rd International Conference on Automata, Languages and Programming (ICALP). Springer-Verlag, Berlin, pp 1-12
12. Girvan M and Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci USA 99(12):7821-7826

13. Gleiser PM and Danon L (2003) Community structure in jazz. Adv Complex Syst 6(04):565-573
14. Guimerà R, Danon L, Díaz-Guilera A, Giralt F and Arenas A (2003) Self-similar community structure in a network of human interactions. Phys Rev E 68(6):065103
15. Hay M, Miklau G, Jensen D, Weis P and Srivastava S (2007) Anonymizing Social Networks. Report, University of Massachusetts Amherst
16. Hay M, Miklau G, Jensen D, Towsley D and Weis P (2008) Resisting structural re-identification in anonymized social networks. Proc VLDB Endow 1(1):102-114
17. Hay M, Li C, Miklau G and Jensen D (2009) Accurate Estimation of the Degree Distribution of Private Networks. In: 9th International Conference on Data Mining (ICDM). IEEE Computer Society, Miami, pp 169-178
18. Herrera-Joancomartí J and Pérez-Solà C (2011) Online Social Honeynets: Trapping Web Crawlers in OSN. In: Proceedings of the 2011 International Conference on Modeling Decisions for Artificial Intelligence (MDAI). Springer, Girona, pp 115-131
19. Lancichinetti A and Fortunato S (2009) Community detection algorithms: a comparative analysis. In: Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools. ICST, Pisa, pp 27:1-27:2
20. Li N, Li T and Venkatasubramanian S (2007) $t$-Closeness: Privacy beyond $k$-anonymity and $l$-diversity. In: 23rd International Conference on Data Engineering (ICDE). IEEE Computer Society, Istanbul, pp 106-115
21. Liu K and Terzi E (2008) Towards identity anonymization on graphs. In: Proceedings of the ACM International Conference on Management of Data (SIGMOD). ACM Press, New York, pp 93-106
22. Lucia, Lo D, Jiang L and Budi A (2012) $kb^e$-Anonymity: Test Data Anonymization for Evolving Programs. In: International Conference on Automated Software Engineering (ASE). ACM Press, New York, pp 262-265
23. Machanavajjhala A, Kifer D, Gehrke J and Venkitasubramaniam M (2007) $l$-diversity: Privacy beyond $k$-anonymity. ACM Trans Knowl Discov Data 1(1):3:1-3:12
24. Mislove A, Marcon M, Gummadi KP, Druschel P and Bhattacharjee B (2007) Measurement and analysis of online social networks. In: Proceedings of the 7th ACM Conference on Internet Measurement (ICM). ACM Press, New York, pp 29-42
25. Newman MEJ and Girvan M (2003) Finding and evaluating community structure in networks. Phys Rev E 69(2):026113
26. Pons P and Latapy M (2005) Computing communities in large networks using random walks. J Graph Algorithms Appl 10(2):191-218
27. Rosvall M and Bergstrom CT (2008) Maps of random walks on complex networks reveal community structure. Proc Natl Acad Sci USA 105(4):1118-1123
28. Sweeney L (2002) $k$-anonymity: a model for protecting privacy. Int J Uncertain Fuzziness Knowl Based Syst 10(5):557-570.
29. Ying X and Wu X (2008) Randomizing Social Networks: a Spectrum Preserving Approach. In: Proceedings of the SIAM International Conference on Data Mining (SDM). SIAM, Atlanta, pp 739-750
30. Ying X, Pan K, Wu X and Guo L (2009) Comparisons of randomization and $k$-degree anonymization schemes for privacy preserving social network publishing. In: Proceedings of the 3rd Workshop on Social Network Mining and Analysis (SNA-KDD). ACM Press, New York, pp 10:1-10:10
31. Zachary WW (1977) An information flow model for conflict and fission in small groups. J Anthropol Res 33(4):452-473
32. Zhang K, Lo D, Lim E and Prasetyo P (2013) Mining indirect antagonistic communities from social interactions. Knowl Inf Syst 35(3):553-583.

33. Zheleva E and Getoor L (2011) Privacy in Social Networks: A Survey. In: C. C. Aggarwal (ed) Social Network Data Analytics, 1st edn. Springer, pp 277-306
34. Zhou B and Pei J (2008) Preserving Privacy in Social Networks Against Neighborhood Attacks. In: Proceedings of the 24th International Conference on Data Engineering (ICDE). IEEE Computer Society, Washington, pp 506-515
35. Zou L, Chen L and Özsu MT (2009) $K$-Automorphism: A General Framework For Privacy Preserving Network Publication. Proc VLDB Endow 2(1):946-957