Privacy- & Utility-Preserving Data Releases over Fragmented Data Using Individual Differential Privacy

Luis Del Vasto-Terrientes[1][a], Sergio Martínez[1][b] and David Sánchez[1,2][c]
[1] *Universitat Rovira i Virgili, Departament d'Enginyeria Informàtica i Matemàtiques,* [2] *CYBERCAT-Center for Cybersecurity Research of Catalonia, Av. Paisos Catalans 26, Tarragona, 43007, Catalonia*
luismiguel.delvasto@fundacio.urv.cat, sergio.martinezl@urv.cat, david.sanchez@urv.cat

Keywords: Data Analysis, Privacy-Preserving Data Release, Individual Differential Privacy, Data Fragmentation.

Abstract: Data fragmentation is the process of splitting data into either attributes or records across multiple databases, thereby improving operational efficiency, minimizing processing requirements, and enhancing data privacy. However, under this approach, data aggregation becomes complex, particularly in environments where adherence to regulatory compliance is essential for organizational data analysis and decision-making tasks. Since the dataset held by each party may contain sensitive information, simply joining local datasets and releasing the aggregated result will inevitably reveal such sensitive information to other parties. Differential Privacy (DP) has become the *de facto* standard for data protection due to its rigorous notion of privacy. However, the strong privacy guarantees it offers result in a deterioration of data utility in several scenarios, such as data releases in either centralized or fragmented data scenarios. This paper explores the application of *Individual Differential Privacy* (iDP) —a formulation of DP conceived to better preserve data utility while still providing strong privacy guarantees to individuals— for data releases in either horizontally or vertically fragmented scenarios. In combination with individual ranking (IR) microaggregation, an iDP-IR privacy-preserving data release system is presented, in which multiple data owners can safely share datasets. Our experiments on two UCI datasets -Adult and Wine Quality- demonstrate that the proposed system for fragmented data can provide robust ε privacy values with reasonable information loss.

1 INTRODUCTION

In the information society, researchers and the general public have been demanding more data, promoting the creation of public repositories such as the Harvard Dataverse, Dryad, and government open data portals. However, these massive data repositories may include both re-identifying and confidential attributes, which can jeopardize the privacy of the individuals they refer. For instance, (Sweeney, 2000) demonstrates that 87% of the U.S. population from the Census 1990 had reported characteristics that likely made them unique by combining ZIP code, date of birth, and gender. Also, (Golle, 2006) shows that 63% of the U.S. population from the Census 2000 can be identified by combining these 3 quasi-identifiers.

Data anonymization has proposed a variety of solutions to protect privacy in data releases (Samarati and Sweeney, 1998; Wang and Xu, 2017). Among these, *Differential Privacy* (DP) has gained popularity across various fields due to its strong privacy guarantees. However, strong privacy comes at the cost of low data utility preservation due to the perturbation applied to data. Despite criticism regarding its usefulness and applicability (Bambauer et al., 2013; Clifton and Tassa, 2013; Blanco-Justicia et al., 2022; J. Domingo-Ferrer, 2021), the data privacy community continues to propose different relaxations and alternative formulations of DP to improve its utility (Dwork et al., 2009; Friedman and Schuster, 2010; Cummings et al., 2024). Some well-known DP relaxations include (ε, δ) (Dwork et al., 2006), Rényi DP (Mironov, 2017), and Zero-Concentrated DP (Bun and Steinke, 2016). These three examples introduce a "small" chance that standard DP guarantees may be broken for the promise of better utility.

An alternative approach to DP relaxation is Individual Differential Privacy (iDP). iDP is a different formulation that allows the data controller to utilize the *actual* knowledge of the dataset —in contrast to the previous DP relaxations mentioned— when

[a] https://orcid.org/0000-0003-0483-8559
[b] https://orcid.org/0000-0002-3941-5348
[c] https://orcid.org/0000-0001-7275-7887

computing the noise addition to generate protected datasets. A notable advantage of iDP is that it provides the strong privacy guarantees of DP to individuals without introducing any risk of breaking these guarantees, while offering better utility. However, solely applying iDP to each attribute value for every record in the dataset would result in low utility due to the high sensitivity each individual value is exposed to. To reduce this sensitivity to noise addition, we employ *individual ranking microaggregation* (IR).

Presented in (Sánchez et al., 2016), DP-IR is a privacy-preserving technique aimed at enhancing utility for data releases by combining standard DP and IR microaggregation. IR replaces detailed data with centroids calculated from clusters of similar values for each attribute in an independent and consecutive manner. Centroids effectively reduce the amount of noise required to achieve DP by decreasing data sensitivity, thereby offering a considerable improvement in utility, which is particularly desirable for data analysis tasks in organizations and industry (Ghazi et al., 2023).

In the literature of DP, its application is commonly assumed in a centralized scenario —called Centralized Differential Privacy (CDP)—, wherein an aggregator collects raw data prior to the application of DP (Yang et al., 2024). However, in many practical applications, datasets are stored in distributed databases —a trend accelerated by the decreasing costs of on-premise infrastructures and cloud computing—. This architecture complicates the previously proposed scenario, as data owners may be unwilling to release raw data, even to a trusted participant.

1.1 Contributions and Plan

In the proposed data release system, each data owner holds an iDP-IR mechanism —a privacy mechanism that combines iDP and IR— to generate ε-iDP datasets that can be safely shared. An aggregator is then in charge of gathering and aggregating protected datasets from different data owners. We develop a protocol for each of the two data fragmentation scenario analyzed in this paper: horizontal and vertical fragmentation. Depending on the data fragmentation scenario, the aggregator must coordinate the ε privacy value and the microaggregation configuration that each data party must locally implement to generate a global ε-iDP.

The implementation of iDP-IR for fragmented data does not only maximizes utility without compromising the use of strong ε values, but is also scalable (no restrictions on the number of data owners) and requires low computational cost in comparison to other approaches that rely on cryptographic techniques commonly inefficient for a large number of data owners.

The paper is organized as follows. First, Section 2 presents a background on distributed privacy-preserving data releases. Section 3 introduces iDP and IR separately and explains how they are combined into iDP-IR to protect sensitive data. Next, Section 4 proposes an honest-but-curious system based on iDP-IR for data releases over fragmented data. Horizontal and vertical fragmentation data scenarios are discussed for the proposed system. Section 5 evaluates the information loss calculation of the global protected dataset generated in the distributed system proposed compared to that of a centralized environment using two UCI datasets: *Adult* and *Wine Quality*. The empirical results obtained are then discussed to analyze the feasibility of applying the proposed iDP-IR-based data release system over fragmented data stored across multiple data owners. Finally, Section 6 presents the conclusions and potential future research directions.

2 RELATED WORK

In a centralized scenario, where data is assumed to be stored in a single repository, a dataset is formalized as follows:

- A is a finite set of attributes a_g, where $g \in \{1,2,...,l\}$.
- R is a finite set of records r_h, where $h \in \{1,2,...,n\}$. Each record r_h is a tuple of values corresponding to each attribute a_g.
- D is a dataset containing the set of records R.

In cases where datasets are fragmented across multiple repositories, we consider two approaches: horizontal fragmentation and vertical fragmentation. Horizontal fragmentation entails storing records with the same attribute schema in multiple databases, whereas vertical fragmentation involves storing different attributes of the same records in multiple databases.

The formalization of datasets considering multiple data owners is as follows:

- A_j is a finite set of attributes a_g, where $g \in \{1,2,...,l\}$.
- R_j is a finite set of records r_h, where $h \in \{1,2,...,n\}$. Each record r_h is a tuple of values corresponding to each attribute $a_g \in A_j$.
- D_j is a dataset containing the set of records R_j. Here, j denotes the specific dataset in which R_j

and A_j are defined.

- For simplicity, $l = |A_j|$ and $n = |R_j|$, where l and n may vary for each j.

Regarding privacy-preserving data releases across data fragmentation scenarios, most previous works have focused on the application of standard DP and Secure Multiparty Computation (SMC) for either horizontal or vertical data fragmentation.

For the horizontal fragmentation scenario, (Alhadidi et al., 2012) presents a model for data releases limited to two-party collaboration only. The algorithm relies on generalizing raw data before applying DP. (Cheng et al., 2020) introduces a differentially private sequential update of Bayesian networks (DP-SUBN), where the parties and a curator collaboratively quantify the correlations of all attribute pairs across all local datasets, which may require significant resources depending on the number of parties and attributes.

For the case of vertically fragmented data, (Mohammed et al., 2014) presents DistDiffGen, the first data release system for this scenario limited to two-party cooperation. It generates an anonymous ε-DP data table tailored for classification tasks. A differentially private latent tree (DPLT) approach is described in (Tang et al., 2021) for solving data publishing in a secure two-party scenario, with an extension to multiparty cases. However, this extension may perform well only when the number of parties is not large. (Wang et al., 2021) presents a semi-honest model called ArbDistDP for two-party collaboration, which privately publishes arbitrarily partitioned data by applying standard ε-DP. This model relies on multiple steps of top-down generalization before noise addition, consuming most of the processing time.

In contrast to the previous works mentioned, our work focuses on utility preservation while providing strong guarantees to individuals by employing iDP. This approach is applicable to both horizontal and vertical data fragmentation, with no limit on the number of data owner participants, and it operates at a low computational cost.

3 INDIVIDUAL DIFFERENTIAL PRIVACY AND MICROAGGREGATION

The DP model (Dwork, 2006) was originally designed for the interactive setting, protecting the results of database queries requesting specific data. In this setting, a sanitizer providing ε-DP outputs sits between the user querying the data and the dataset. However, its application was extended to other contexts such as data releases, differentially private machine learning models, data collection, among others. In this work we focus on data releases at microdata level, which is information at the level of individual respondents.

3.1 Differential privacy

DP has gained popularity among the scientific community due to its strong privacy guarantee, which ensures that the presence or absence of an individual in the dataset has little effect on the output. In Definition 1, DP is formally defined.

Definition 1. *A differential privacy (DP) mechanism* ν *gives* ε*-DP if, for all datasets* D_1 *and* D_2 *differing in at most one record (i.e., neighbor datasets) and all* $S \subset$ *Range(*ν*), we have:*

$$Pr(\nu(D_1) \in S) \leq exp(\varepsilon) Pr(\nu(D_2) \in S) \qquad (1)$$

The privacy parameter ε, also known as privacy budget, determines the level of disclosure the system will tolerate. A smaller budget corresponds to stronger privacy and consequently, less data utility. According to (Dwork, 2008), values in the range $\varepsilon = [0.1, 1]$ provide robust privacy guarantees.

The amount of noise added to the original values is proportional to the sensitivity of those values to modifications. This noise represents how much the value may be distorted in the output. The global sensitivity is defined as the largest difference in variability between neighboring datasets within the same domain $\mathcal{D}$, and it is formalized as follows:

Definition 2. *Let u be a positive integer and* $\mathcal{D}$ *be a collection of datasets containing* D_1 *and* D_2*. The global sensitivity of a function* $f : \mathcal{D} \rightarrow R^u$ *is*

$$\triangle f = \max_{\substack{D_1, D_2 \in \mathcal{D} \\ d(D_1, D_2)=1}} ||f(D_1) - f(D_2)||_1, \qquad (2)$$

where $d(D_1, D_2)$=1 represents that datasets D_1 and D_2 differ in one record.

Assuming that (r_h, a_g) is a real value response located in a dataset D, in record h and attribute g, for a certain query Q that must be masked by adding random noise $Y(D)$, a randomized response $\nu(D) = Q_{(r_h, a_g)}(D) + Y(D)$ is computed.

In the literature, the most common method applied to attain DP is the Laplace mechanism, which generates random noise using the Laplace distribution. By controlling the scale of the noise, the Laplace mechanism can balance the trade-off between privacy and utility, making it a versatile and widely-used technique in the context of DP.

3.2 Individual Differential Privacy and Microaggregation

iDP (Soria-Comas et al., 2017) is a privacy model that can incur less information loss than standard DP, while giving individuals the same privacy protection as DP. In contrast to standard DP, iDP utilizes knowledge from the *actual* dataset containing the individuals to protect, while DP follows the worst case scenario, i.e., it is designed to provide robust privacy protection under the most challenging and adversarial conditions.

iDP is formalized as follows:

Definition 3. *Given a dataset D_1, an iDP mechanism ν gives ε-iDP if, for any dataset D_2 that is neighbor of D_1, and any $S \subset Range(\nu)$, we have:*

$$\begin{aligned} exp(-\varepsilon)Pr(\nu(D_2) \in S) \leq Pr(\nu(D_1) \in S) \\ \leq exp(\varepsilon)Pr(\nu(D_2) \in S) \end{aligned} \quad (3)$$

For data releases, iDP can be attained by calibrating the noise addition to local sensitivity instead of global sensitivity. This results in better preservation of utility because it typically requires much smaller noise addition than global sensitivity, where the actual dataset specifications are not considered.

The local sensitivity is defined as the largest difference in variability between the actual dataset D_1 and any neighbor D_2 within the domain $\mathcal{D}$, and it is formalized as follows:

Definition 4. *Let u be a positive integer and $\mathcal{D}$ is a collection of datasets containing D_1 as the actual dataset and D_2 any neighbor dataset. The local sensitivity of a function $LS_f : \mathcal{D} \rightarrow R^u$ is*

$$LS_f = \max_{\substack{D_2 \in \mathcal{D} \\ d(D_1,D_2)=1}} ||f(D_1) - f(D_2)||_1, \quad (4)$$

One of the most interesting properties of iDP (and DP) is composability, which we develop in this work to generate integrated ε-iDP datasets in data fragmentation scenarios. Composability is considered for a sequence of iDP mechanisms, such as consecutive queries executed over the sanitizer.

Theorem 1. *Sequential composition: Let ν_1 and ν_2 be randomized functions accessing non-disjoint datasets D_1 and D_2, satisfying ε_1-iDP and ε_2-iDP respectively, the combined output satisfies (ε_1 + ε_2)-iDP.*

Theorem 2. *Parallel composition: Let ν_1 and ν_2 be randomized functions accessing disjoint datasets D_1 and D_2, satisfying ε_1-iDP and ε_2-iDP respectively, the combined output satisfies $\max(\varepsilon_1, \varepsilon_2)$-iDP.*

The composition property of the iDP model is crucial in this study to effectively guarantee a global ε-iDP data release output. Depending on the type of data fragmentation, an aggregator must accurately assign the privacy budget that each data owners needs to implement. Further elaboration on this process is provided in Section 4.2.

iDP was originally designed for the interactive setting, same as DP. In this scenario, a user poses a query over a dataset to receive an answer in the form of aggregated data (e.g., average and medians). For microdata-level data releases over distributed data using iDP, we leverage on a pre-processing step based on data microaggregation to provide greater data utility without comprising strong privacy.

3.3 Individual Ranking Microaggregation

In data releases, to generate an ε-iDP record r_h, by sequential composition (see Theorem 1), it is required to add ε/l-iDP $\forall g$, (r_h, a_g), where l is the number of attributes in D. Thus, for h=1 the consecutive set of queries $\forall g$, $Q_{(r_1,a_g)}(D)$ results in an ε-individual differentially private record r_1. Such queries are very sensitive and require a considerable amount of ε to provide reasonable utility, even if we consider local sensitivity rather than global sensitivity.

To reduce the sensitivity, this work applies the mechanism proposed in (Sánchez et al., 2016), which relies on a microaggregation-based approach called individual ranking (IR) as a pre-processing step.

Microaggregation is a perturbative masking technique where the attribute values of each record are replaced by aggregating values of similar records. It encompasses two methods: univariate and multivariate microaggregation. The univariate method - approach known as IR- aggregates the values of each attribute one at a time in a consecutive and independent manner, while multivariate methods aggregate all attributes simultaneously, making the latter computationally more costly.

IR clusters a dataset D_j so that values for each attribute $a_g \in A_j$ are sorted either in increasing or decreasing order for subsequent aggregation into successive clusters of k elements. From each of the clusters formed, a centroid is generated, typically through a simple arithmetic average. This centroid then replaces the original values belonging to the cluster. IR produces low utility loss but a high disclosure risk when applied in isolation. However, this is compensated by the less sensitive centroids it produces compared to the original values.

The low computational cost required by IR is op-

timal for enhancing data utility in combination with iDP for fragmented data in a distributed environment, where, instead of relying on one powerful server specification, the workload is commonly shared across several less powerful parties.

The IR microaggregation method is explained in the following algorithm, assuming that the computation is performed by a single data owner.

Data: Dataset D, Parameter value k
Return: IR dataset D^*
$D^* \leftarrow D$;
{Copy of the original dataset to microaggregate}
$numOfClusters \leftarrow floor(n/k)$;
for all $a_g \in A^*$ **do**
  $D^* \leftarrow$ ascendingSortForAttribute(D^*, a_g);
  for $cluster \leftarrow 1$ **to** $numOfClusters$ **do**
    $partialCluster, centroid \leftarrow$ null;
    {Cluster and centroid}
    $partialCluster \leftarrow$ computeCluster($cluster$, D^*, a_g, k);
    $centroid \leftarrow$ calculateCentroid($partialCluster$);
    replaceValuesByCentroid(D^*, $partialCluster$, $centroid$);
  end for
end for
reorderRecordsToOriginalOrder(D, D^*)

Algorithm 1: Individual Ranking Microaggregation

The algorithm presented receives dataset D and the parameter value k as inputs to compute.

First, a copy of D, called D^*, is created. D^* is then the dataset to be microaggregated and returned once the process is completed. The number of clusters generated depends on the number of records n and the value k. The function $floor$ is used to return the largest integer that is less than or equal to the argument, so that for $floor(102/5)$ the function returns 20. In this example, 20 clusters are computed and the remaining values are grouped within the last cluster.

Continuing with Algorithm 1, for each attribute $a_g \in A^*$, the values are sorted in an ascending way, as well as the rest of record values. Next, the algorithm continues with the clustering and centroid calculations for each attribute a_g independently. $partialCluster$ groups the set of values that belongs to $cluster$, which ranges between 1 and $numOfClusters$. Function $generateCluster$ calculates and returns at least k similar values from attribute $a_g \in D^*$, forming the cluster.

$generateCluster$ returns an array of values, from which the original values are mapped and replaced by the computed centroid. Once all centroids are calculated for all $a_g \in A^*$, Algorithm 1 reorders each record to its original position as in the dataset D, resulting in the microaggregated dataset D^*.

4 PRIVACY-PRESERVING DATA RELEASES FOR FRAGMENTED DATA APPLYING IDP-IR

This section aims to introduce an honest-but-curious iDP-IR-based system model and its associated protocols for data releases. The model considers distributed data in either horizontal or vertical fragmentation and must be locally protected prior to being shared with a trusted party to guarantee privacy. Also, it ensures that each participant in the system runs the protocol exactly as specified (no deviations or malicious parties).

4.1 Distributed system model

Let us consider an honest-but-curious distributed system where several data owners collaborate to construct a robust private dataset for analysis without compromising data privacy. In this architecture, it is assumed that both the aggregator and the participating parties adhere to the protocol.

Figure 1 depicts the architecture of the system and illustrates the participants and the basic communication flow.

The proposed architecture encompasses the following participants:

- Data owners ϕ: Consider the set of data owners ϕ_j, where $j \in \{1,2,...m\}$. These data owners may involve on-premise databases as well as cloud databases holding datasets D_j. Each ϕ_j possesses an iDP-IR mechanism ν_j that generates an ε-individual differentially private dataset D'_j. This work assumes that $|\phi| \geq 2$.
- Aggregator κ: It is in charge of orchestrating the aggregation of ε-individual differentially private datasets D'_j into a combined protected dataset D', so that $D' = D'_1 \cup D'_2 \cup ... \cup D'_m$. This work assumes that $|\kappa| = 1$.
- Users ρ: External users represent individuals who are willing to use the released data for analysis. Their participation in the model may not be mandatory, meaning that data releases may be shared only among data owners to enhance their knowledge for decision-making.

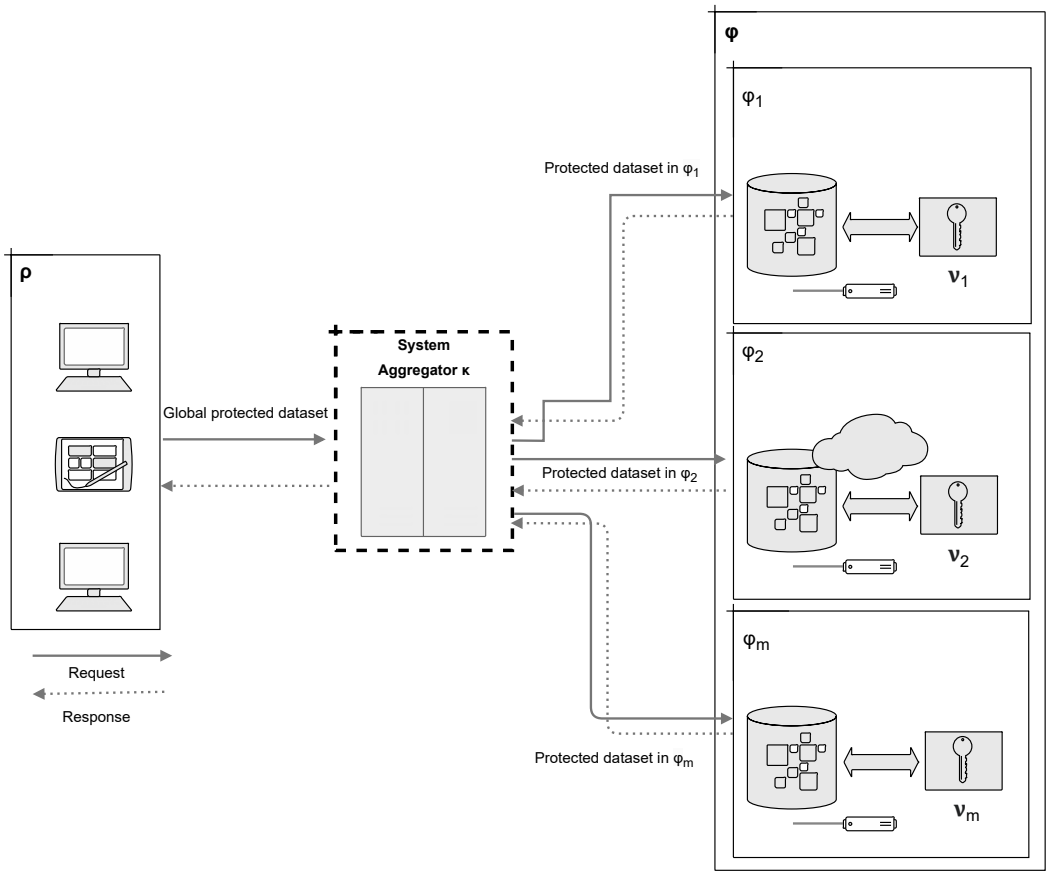


Figure 1: Architecture of the proposed model

In the proposed system, the aggregator sets the privacy parameters to be applied by each data owner and validates the data schema between parties. Each data owner is equipped with an iDP-IR mechanism. Initially, the mechanism computes IR microaggregation over D_j —which represents the dataset each data owner is willing to share— thereby generating a microaggregated dataset D_j^*.

Finally, for each centroid computed for the several clusters generated in D_j^*, ε-iDP is applied. The noisy centroid value obtained then replaces the values of the cluster associated with the centroid in the microaggregated dataset, resulting in an iDP-IR dataset D_j'. This dataset is sent to the aggregator, who is responsible for enforcing the protocol for delivering all data owners' protected datasets before generating a global protected dataset for data release. This work assumes a secure channel connects all parties involved in the model.

4.2 Composition property for data fragmentation & iDP-IR

We now explore two possible scenarios of privacy-preserving data releases in fragmented data: horizontal and vertical fragmentation. Depending on the scenario, the communication flow and data exchange between ϕ and κ differ.

Let the global dataset D be distributed between data owners ϕ in either an horizontal (Joyce and Nirmalrani, 2015; Sauer and Hao, 2015) or vertical way (Vaidya, 2008). In an horizontally fragmented data, two data owners ϕ_1 and ϕ_2 contain a disjoint set of individuals with the same schema, so that $A_1 = A_2$ and $R_1 \cap R_2 = \oslash$. These parties are willing to cooperate by combining samples from multiple sources to obtain enhanced datasets for robust data analysis. For instance, two or more small marketing companies holding different datasets with the same attributes may desire to gain knowledge by collecting different insights.

On the other hand, vertical fragmented data collects different sets of attributes about the same individuals stored in multiple data owners ϕ_j. For two data owners ϕ_1 and ϕ_2, $A_1 \cap A_2 = \oslash$ and $R_1 = R_2$. As an example, two subsidiary companies belonging to the same parent company may have stored different features about the same clients. The parent company may decide to join these datasets to perform descriptive and predictive analyses for decision-making.

In the context of these two data fragmentation scenarios, an aggregated ε-iDP dataset in κ may be achieved for data release by correctly applying the composition property as described in Theorem 1 and Theorem 2. This process is explained and illustrated in the following sections, along with the protocols between the participants for each data fragmentation scenario.

4.2.1 Horizontal fragmentation

For horizontally fragmented data, parallel composition is utilized to obtain an ε-individual differentially private dataset D' from protected datasets D_j'. Following Theorem 2, if $(\varepsilon_1 = \varepsilon_2 = ... = \varepsilon_m)$, an ε-iDP dataset D' is computed by ensuring that each ν_j provides ε-iDP records $\in D_j'$ accordingly.

Protocol for Horizontal Fragmentation

1. κ requests $\forall j \in A_j$, i.e., the attributes each data owner ϕ_j is willing to share.
2. $\forall j \in \phi_j$, A_j metadata is sent to κ.
3. κ verifies if $A_1 = A_2 = ... = A_m$.
4. κ requests $\forall j \in \phi_j$ a protected dataset D_j' with parameters $k \geq 3$ and equal ε-iDP, following Theorem 2.
5. $\forall j \in \phi_j$, k partitions are computed $\forall g$, $a_g \in A_j$, generating a microaggregated dataset D_j^*.
6. $\forall j \in \phi_j$, ε-iDP is applied over D_j^* by adding noise to each centroid, which then replaces the attribute values in the cluster, generating a protected dataset D_j'.
7. κ retrieves $\forall j \in D_j'$, aggregating them into a protected dataset D'.
8. κ releases D' to ρ and ϕ.

4.2.2 Vertical fragmentation

For the case of vertical data fragmentation (i.e., disjoint attributes), where different mechanisms ν_j are independently computing different sets of attribute outputs of the same individual, let us consider Theorem 1 to compute an ε-iDP-IR private dataset D'.

Assuming that the aggregator equitably divides ε between all data owners ϕ_j, then a $(\sum_{\phi_j=1}^{m} \varepsilon_j)$-iDP-IR dataset D' is obtained from the combination of their individual outputs. Locally in ϕ_j, each masked attribute value output is computed as $\frac{\varepsilon_j}{l}$, resulting in an iDP record $(\sum_{a_g=1}^{l} \varepsilon_j)$.

In vertical fragmentation, either all data owners must share a common identifier or the data must be fragmented in a way that allows for record reconstruction during the defragmentation process at the aggregator. As a result, this fragmentation method is computationally more demanding than horizontal fragmentation.

Protocol for Vertical Fragmentation

1. κ requests $\forall j \in A_j$, i.e., the attributes each data owner ϕ_j is willing to share.
2. $\forall j \in \phi_j$, A_j metadata is sent to κ.
3. κ verifies if $A_1 \cap A_2 \cap ... \cap A_m = \oslash$.
4. κ requests $\forall j \in \phi_j$ a protected training dataset D'_j with parameters $k \geq 3$ and $\frac{\varepsilon}{m}$-iDP, following Theorem 1.
5. $\forall j \in \phi_j$, k partitions are computed $\forall g$, $a_g \in A_j$, generating a microaggregated dataset D^*_j.
6. $\forall j \in \phi_j$, $\frac{\varepsilon}{m}$-iDP is applied over D^*_j by adding noise to each centroid, which then replaces the attribute values in the cluster, generating a protected dataset D'_j.
7. κ retrieves $\forall j \in D'_j$, constructs records and aggregates them into a protected dataset D'.
8. κ releases D' to ρ and ϕ.

5 EXPERIMENTAL EVALUATION

This section evaluates the iDP-IR-based privacy-preserving data release system proposed for fragmented data. The focus is on assessing the information loss obtained from the protected datasets using iDP-IR in a centralized environment with different (ε, k) pairs.

This is compared to the information loss obtained from the aggregated protected datasets following the horizontal and vertical fragmentation protocols presented, using the same (ε, k) pairs.

The tests were conducted on a Windows 11 Home PC with an Intel i7-1355U CPU @5.00 GHz and 16 GB DDR5 RAM.

5.1 Evaluation datasets

The following UCI datasets have been selected for the experimental evaluation due to their diverse attribute data types, ensuring a more robust analysis across different scenarios:

- *Adult* is a well-known dataset available on the UCI Machine Learning Repository, comprising 48,842 records of census income information. Its objective is to predict whether income exceeds $50,000 per year. For the *Adult* dataset evaluation, the following attributes are used: age, workclass, education, marital status, occupation, relationship, race, sex, hours per week, and native country. The attributes in this dataset include discrete numerical values and categorical data.

  A pre-processing step was carried out to remove records with missing values. The final dataset used in our evaluation consists of 45,222 records.

- The *Wine Quality* dataset (Cortez et al., 2009) contains Portuguese red and white wine samples, collected from protected designation of origin samples that were tested at the official certification entity from May 2004 to February 2007.

  The dataset originally split into two separate datasets, *Red* and *White* wine. In our study, we combined the two samples into a single dataset containing 6,497 records. The following attributes are used: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol. This dataset includes continuous and discrete numerical attributes.

The evaluation consists of comparing the information loss between the protected dataset D_{base} —which results from applying the data masking process to the original *Adult* and *Wine Quality* datasets using iDP-IR in a centralized manner— and the results obtained from the aggregated protected dataset D' using the presented protocols. In this centralized scenario, we assume a single party holds the entire datasets and an iDP-IR mechanism.

Given the composition property of iDP, we expect a comparable level of information loss between the global protected dataset D', result of the aggregation of protected datasets subsets D'_j, and D_{base}.

To quantify the information loss of the anonymized dataset concerning D_{base} and D', we used the Sum of Squared Errors (SSE) in our tests. SSE represents the sum of squared distances between the original and anonymized dataset records. In order to show a more informative measure, the mean SSE is computed by dividing the SSE by the number of the records in the dataset.

$$MeanSSE = (1/n) \sum_{h=1,\ldots,n} dist(r_h, r'_h)^2 \qquad (5)$$

To calculate the distance between the original record r_h and the anonymized record r'_h, we considered the average of the distances between attributes. Then, for two records r_h and r'_h we have

$$dist(r_h, r'_h) = (1/l)\sqrt{\begin{array}{r}(d_1((r_h, a_1), (r'_h, a'_1))/\sigma_1^2)^2 + \\ \ldots + (d_l((r_h, a_l), (r'_h, a'_l))/\sigma_l^2)^2\end{array}} \qquad (6)$$

where d_g is the distance between values of attribute a_g, σ_g^2 is the sample variance of attribute a_g

in the original dataset and l is the number of attributes in D_j.

Centroids computation for categorical attributes For categorical attributes, we sort the attribute values based on their frequency of appearance. Clusters are then formed by selecting k contiguous values. For example, assuming the frequencies of a_1 = 10 and a_2 = 15, we have a_2^1 and a_1^2, where the superscripts indicate the order. For a cluster of 3 elements, an example may be $\{ a_2^1, a_2^1, a_1^2 \}$. Since it is not possible to operate directly on the values, we operate on the indices instead.

The centroid of a cluster is the value corresponding to the average index. From the previous example, $\frac{1+1+2}{3} = 1.33$. Notice that the centroid does not correspond to an actual value. To convert it into an actual value, we round the index to the nearest integer (i.e., 1 in this example). However, this rounding step is performed only after noise has been added.

5.1.1 Baseline - Centralized Scenario

The mean SSE obtained for the centralized scenario -used as a baseline for comparison- for the *Adult* and *Wine Quality* datasets are shown in Figure 2 and Figure 3, respectively. These figures present a heatmap representing the mean SSE calculated for different (ε, k) pairs, where each attribute value is the result of the average of fifty (50) consecutive runs of iDP-IR.

The color degradation in the heatmaps from cream to red indicates that, the redder the cell, the larger the information loss between the original and anonymized datasets. Notice that we included the mean SSE obtained for $k = 1$, which is equivalent to applying plain iDP over each attribute value, i.e., no microaggregation step.

Notice that the privacy budgets are set to $\varepsilon = [0.1, 1]$, a range commonly used in the literature for robust privacy protection. Because of the difference in the number of records between the two datasets analyzed, the k values are set to $[50, 2000]$ and $[50, 300]$ for the *Adult* and *Wine Quality* datasets, respectively.

Theoretically, for larger ε, we may expect a decrease in SSE due to the relaxation of the privacy budget, resulting in less noise injection. Also, for larger k, we may also expect a decrease in SSE because of the reduced data sensitivity, resulting in less required noise. This pattern is exactly what is observed in Figures 2 and 3.

These results now serve as the baseline for evaluating the proposed iDP-IR system model. The data fragmentation configuration of the datasets is described in detail for both scenarios, outlining how the

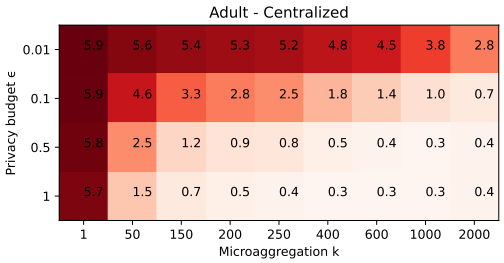


Figure 2: Adult iDP-IR SSE Baseline

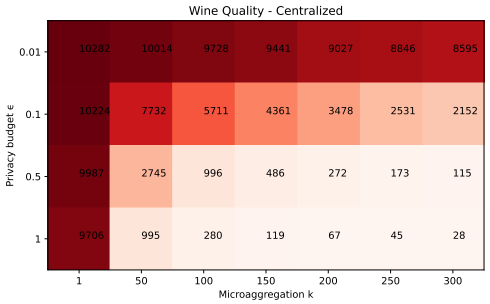


Figure 3: Wine Quality iDP-IR SSE Baseline

datasets are partitioned and distributed among multiple predefined parties.

Before evaluating the D_{base} and the data fragmentation scenarios, we illustrate the advantages that iDP-IR offers in improving information loss over plain iDP to each attribute value and DP-IR, which applies IR microaggregation and DP rather than iDP (i.e., global sensitivity).

Evaluation of iDP and iDP-IR The application of IR as a pre-processing step significantly impacts the reduction of information loss in data releases. This effect is depicted in Figures 2 and 3. Specifically, when considering plain iDP, i.e., $k = 1$, the results show that the decrease in information loss is minimal while increasing the ε values.

This observation indicates that, even though local sensitivity is employed, each attribute value independently maintains a high sensitivity to noise modifications. Consequently, achieving meaningful utility necessitates higher ε values. However, this requirement inherently results in a trade-off where the utility gains are offset by a corresponding decrease in privacy protection.

Evaluation of DP-IR and iDP-IR We now compare the information loss obtained from comparing iDP-IR (local sensitivity) and DP-IR (global sensitivity), the latter as presented in (Sánchez et al., 2016).

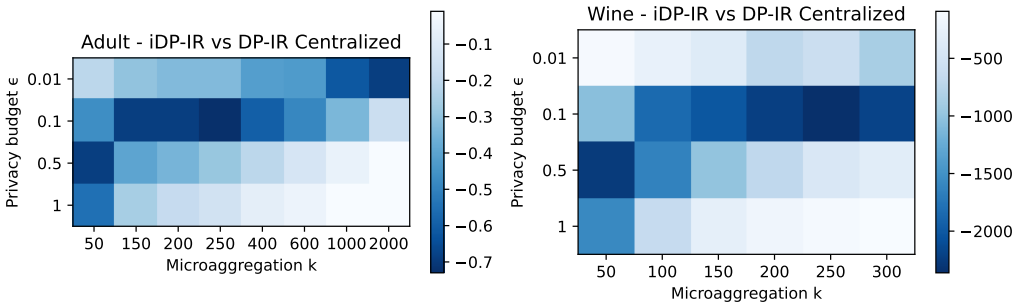


Figure 4: Adult (left) and Wine Quality (right) Comparison of iDP-IR and DP-IR

Figure 4 shows the differences obtained from iDP-IR and DP-IR for both *Adult* (left) and *Wine Quality* (right). The values obtained are the difference between the mean SSE obtained in Figures 2 and 3 for iDP-IR and those obtained using DP-IR. Notice that, for all pairs (ε, k), iDP-IR obtains better SSE compared to DP-IR, i.e., lower values from the former result in a negative SSE relative to the latter.

5.1.2 Horizontally fragmented datasets configuration

Adult The evaluation considers four data owners storing randomized subsets of the dataset in an unbalanced split: ϕ_1 contains 11,100 records, ϕ_2 contains 15,000 records, ϕ_3 contains 12,000 records, and ϕ_4 contains 7,122 records.

Wine Quality The evaluation considers three data owners storing randomized subsets of *Wine Quality* in an unbalanced split: ϕ_1 contains 500 records, ϕ_2 contains 1,000 records, and ϕ_3 contains 4,997 records.

In addition to this unbalanced distribution, we implement a shuffling mechanism that allows records to be exchanged among the different data owners during each iteration of the evaluation process. This shuffling works as a tool for examining how the redistribution of records influences the performance of the information loss. Hence, we can investigate the robustness of the iDP-IR system proposed and better understand the implications of data variability on the outcomes of the evaluation.

For the evaluation, the domain of the attributes in this dataset has been limited to twice the maximum attribute value.

5.1.3 Vertically fragmented datasets configuration

Adult We considered ten (10) data owners, each storing the same number of records belonging to the same individuals but with different attributes. Therefore, each data owner holds a single attribute, such that ϕ_1 contains attribute age, ϕ_2 contains attribute workclass, ..., ϕ_{10} contains native country.

Wine Quality The dataset's attributes are distributed across three (3) data owners, grouping individuals' attribute values as follows: ϕ_1 contains attributes { fixed acidity, volatile acidity, citric acid }, ϕ_2 contains { residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide }, and ϕ_3 contains { density, pH, sulphates, alcohol }.

As a pre-processing step, each record holds an identifier prior to the split, allowing for the defragmentation of the records when κ receives the subsets of protected data from the data owners ϕ_j.

5.1.4 Evaluation results

Figures 5 and 6 present the results obtained for the two datasets by comparing the average mean SSE from the data fragmentation configurations and the baseline (i.e., centralized scenario) for pairs (ε, k). Hence, each cell in the heatmap is computed from the difference between the mean SSE baseline averages (as presented in Figures 2 and 3) and the results obtained from the data fragmentation scenarios.

The creamer the cell, the closer the difference between the mean SSE of the centralized and data fragmentation scenarios is to 0. Mean SSE lower than computed on the baseline is also set to 0 (i.e., cream cell). The redder the cell, the larger the information loss between the data fragmentation and the baseline.

Notice that the mean SSE for the data fragmentation configurations is also calculated as the average after fifty (50) runs.

Adult Overall, the differences between the mean SSE averages for the horizontally fragmented data scenarios are minimal compared to the baseline for low k values (i.e., $k = [50, 250]$). However, some information loss can be observed for larger k values, where it increases compared to the baseline. This behavior may occur for data owners with a lower number of records to share because microaggregation depends on data distribution, data homogeneity, and group size, which can result in less optimal data clustering for subsets of records. In this particular case, considering that *Adult* mostly contains categorical attributes (a type that has no natural order and has been artificially sorted based on data distribution), the results are prone to larger differences with respect to the baseline, as the sorted values depend on the actual distribution on each data owner.

For vertical fragmentation, the variation with respect to the baseline is negligible, with the maximum differences obtained being around 0.04, as shown in Figure 5. Parameter k does not significantly affect the resulting outcome in this scenario because the entire set of values for each attribute remains the same as in the baseline. This occurs because IR microaggregation treats each attribute independently. Hence, a correct implementation of the iDP-IR composition property for this type of fragmentation results in similar information loss compared to the centralized baseline.

Wine Quality For the *Wine Quality* dataset, the horizontal fragmentation scenario provides robust results, similar to that of the vertical fragmentation, according to the similar differences between both and the centralized outputs. Notice that for this dataset, most of the cells in the heatmap are cream. As such, the results are either as good as in the centralized scenario or yield slightly better results due to the unpredictable nature of the noise addition. For $\varepsilon = \{0.01, 0.1\}$, where the mean SSE in the centralized scenario is in the thousands, the difference with respect to both data fragmentation scenarios are below 300. For $\varepsilon = \{0.5, 1\}$, we mostly observe slightly different mean SSE values, indicating that both horizontal and vertical splitting does not significantly impact IR clustering for this dataset.

6 CONCLUSIONS

This work proposes a distributed data release model in which a set of parties is willing to release a protected dataset to an aggregator. Following an honest-but-curious model, data owners may only share data using iDP-IR as a utility-preserving privacy mechanism, which relies on the application of IR microaggregation as a pre-processing step to iDP, thereby reducing data sensitivity to noise injection. Consequently, a lesser amount of noise injection is required to achieve ε-iDP datasets.

The results from the experimental scenarios for both horizontal and vertical fragmentation data show that it is possible to achieve similar information loss from distributed iDP-protected datasets to those obtained in a centralized setting for the same pairs (ε, k). This is achieved thanks to the composition properties that iDP inherits from standard DP.

ACKNOWLEDGEMENTS

This research was funded by the European Commission (project H2020-871042 "SoBigData++"), the Government of Catalonia (ICREA Acadèmia Prize to D. Sánchez), MCIN/AEI/ 10.13039/501100011033 and "ERDF A way of making Europe" under grant PID2021-123637NB-I00 "CURLING", and the EU's NextGenerationEU/PRTR via INCIBE (project "HERMES" and INCIBE-URV cybersecurity chair).

REFERENCES

Alhadidi, D., Mohammed, N., Fung, B., and Debbabi, M. (2012). Secure distributed framework for achieving ε-differential privacy. In Fischer-Hübner, S. and Wright, M., editors, *Privacy Enhancing Technologies*, pages 120–139, Berlin, Heidelberg. Springer Berlin Heidelberg.

Bambauer, J., Muralidhar, K., and Sarathy, R. (2013). Fool's Gold: An Illustrated Critique of Differential Privacy. Arizona Legal Studies Discussion Paper No. 13-47.

Blanco-Justicia, A., Sánchez, D., Domingo-Ferrer, J., and Muralidhar, K. (2022). A critical review on the use (and misuse) of differential privacy in machine learning. *ACM Comput. Surv.*, 55(8).

Bun, M. and Steinke, T. (2016). Concentrated differential privacy: Simplifications, extensions, and lower bounds. http://arxiv.org/abs/1605.02065.

Cheng, X., Tang, P., Su, S., Chen, R., Wu, Z., and Zhu, B. (2020). Multi-party high-dimensional data publishing under differential privacy. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1557–1571.

Clifton, C. and Tassa, T. (2013). On syntactic anonymity and differential privacy. In *2013 IEEE 29th International Conference on Data Engineering Workshops (ICDEW)*, pages 88–93. IEEE.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547 – 553.

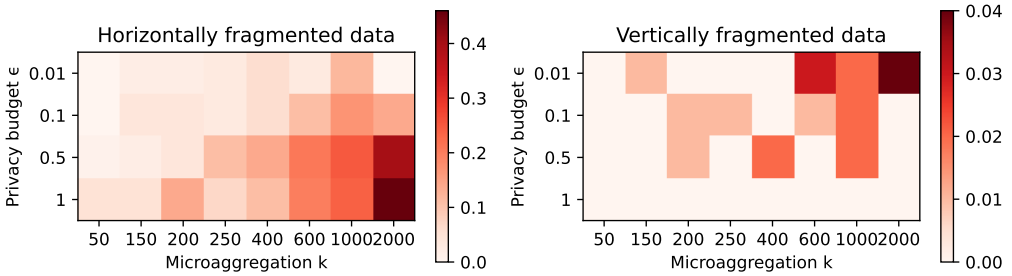


Figure 5: Adult SSE comparison between the centralized scenario and the data fragmentation scenarios

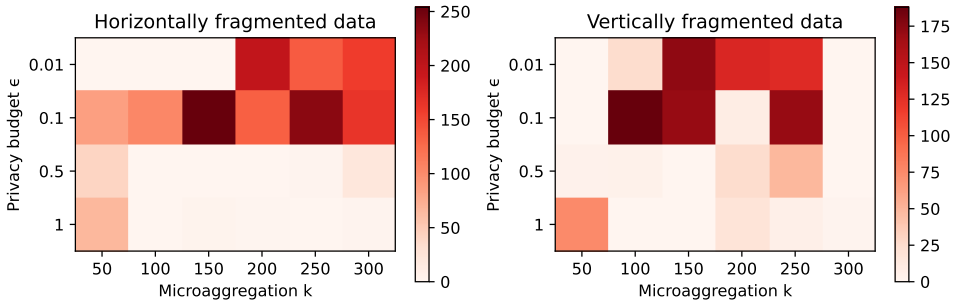


Figure 6: Wine Quality SSE comparison between the centralized scenario and the data fragmentation scenarios

Cummings, R., Desfontaines, D., Evans, D., Geambasu, R., Huang, Y., Jagielski, M., Kairouz, P., Kamath, G., Oh, S., Ohrimenko, O., Papernot, N., Rogers, R., Shen, M., Song, S., Su, W., Terzis, A., Thakurta, A., Vassilvitskii, S., Wang, Y.-X., Xiong, L., Yekhanin, S., Yu, D., Zhang, H., and Zhang, W. (2024). Advancing Differential Privacy: Where We Are Now and Future Directions for Real-World Deployment. *Harvard Data Science Review*, 6(1). https://hdsr.mitpress.mit.edu/pub/sl9we8gh.

Dwork, C. (2006). *Differential Privacy*, pages 1–12. Springer Berlin Heidelberg, Berlin, Heidelberg.

Dwork, C. (2008). *Differential Privacy: A Survey of Results*, pages 1–19. Springer Berlin Heidelberg, Berlin, Heidelberg.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. (2006). Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 486–503. Springer.

Dwork, C., Naor, M., Reingold, O., Rothblum, G. N., and Vadhan, S. (2009). On the complexity of differentially private data release: efficient algorithms and hardness results. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, page 381–390, New York, NY, USA. Association for Computing Machinery.

Friedman, A. and Schuster, A. (2010). Data mining with differential privacy. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 493–502, New York, NY, USA. ACM.

Ghazi, B., Hu, X., Kumar, R., and Manurangsi, P. (2023). Differentially private data release over multiple tables. In *Proceedings of the 42nd ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS '23, page 207–219, New York, NY, USA. Association for Computing Machinery.

Golle, P. (2006). Revisiting the uniqueness of simple demographics in the us population. In *Proceedings of the 5th ACM Workshop on Privacy in Electronic Society*, WPES '06, pages 77–80, New York, NY, USA. ACM.

J. Domingo-Ferrer, D. Sánchez, A. B.-J. (2021). The limits of differential privacy (and its misuse in data release and machine learning). *Communications of the ACM*, 64(7):33–35.

Joyce, M. S. and Nirmalrani, V. (2015). Privacy in horizontally distributed databases based on association rules. In *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, pages 1–6.

Mironov, I. (2017). Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275.

Mohammed, N., Alhadidi, D., Fung, B. C., and Debbabi, M. (2014). Secure two-party differentially private data release for vertically partitioned data. *IEEE Transactions on Dependable and Secure Computing*, 11(1):59–71.

Samarati, P. and Sweeney, L. (1998). Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression.

Sánchez, D., Domingo-Ferrer, J., Martínez, S., and Soria-Comas, J. (2016). Utility-preserving differentially private data releases via individual ranking microaggregation. *Information Fusion*, 30:1 – 14.

Sauer, B. and Hao, W. (2015). Horizontal cloud database partitioning with data mining techniques. In *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, pages 796–801.

Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D., and Megías, D. (2017). Individual differential privacy: A utility-preserving formulation of differential privacy guarantees. *IEEE Transactions on Information Forensics and Security*, 12(6):1418–1429.

Sweeney, L. (2000). Simple demographics often identify people uniquely. *Health (San Francisco)*, 671(2000):1–34.

Tang, P., Cheng, X., Su, S., Chen, R., and Shao, H. (2021). Differentially private publication of vertically partitioned data. *IEEE Transactions on Dependable and Secure Computing*, 18(2):780–795.

Vaidya, J. (2008). *A Survey of Privacy-Preserving Methods Across Vertically Partitioned Data*, pages 337–358. Springer US, Boston, MA.

Wang, H. and Xu, Z. (2017). Cts-dp: Publishing correlated time-series data via differential privacy. *Knowledge-Based Systems*, 122:167 – 179.

Wang, R., Fung, B. C., Zhu, Y., and Peng, Q. (2021). Differentially private data publishing for arbitrarily partitioned data. *Information Sciences*, 553:247–265.

Yang, M., Guo, T., Zhu, T., Tjuawinata, I., Zhao, J., and Lam, K. (2024). Local differential privacy and its applications: A comprehensive survey. *Computer Standards & Interfaces*, 89:103827.