Automation and Machine Learning (2023)
Clausius Scientific Press, Canada

DOI: 10.23977/autml.2023.040205
ISSN 2516-5003 Vol. 4 Num. 2

Perturbation Methods for Protecting Data Privacy: A Review of Techniques and Applications

Safiye Turgay[1,a,*], İlker İlter[1,b]

[1]Department of Industrial Engineering, Sakarya University, Sakarya, Turkey
[a]safiyeturgay2000@yahoo.com, [b]ilkerilter54@hotmail.com
[*]Corresponding author

Keywords: Privacy; Data mining protecting privacy; Big data; Multi-dimensional grid; Privacy enhancement; Perturbation approach

Abstract: Perturbation methods are mathematical techniques used to add controlled noise or randomness to data to protect privacy while allowing data analysis. Various methods, such as randomized response, differential privacy, secure multi-party computation, noise addition, and sampling and aggregation, are used to protect sensitive information from disclosure or exploitation. These methods have been successfully applied in machine learning, statistics, and cryptography to ensure data privacy. However, their implementation must be carefully designed to avoid compromising data accuracy or introducing bias in analysis. Mostly, perturbation methods offer a promising approach to protect data privacy in various fields. This review provides an overview of perturbation methods used to protect data privacy in various fields, including machine learning, statistics, and cryptography. Perturbation methods involve adding controlled noise or randomness to data to preserve privacy while still allowing data analysis.

1. Introduction

The increasing amount of digital data generated by individuals and organizations has raised concerns about the privacy and security of sensitive information. Unauthorized access to personal data can lead to identity theft, financial fraud, and other malicious activities. Data privacy is a critical issue, particularly for sensitive data that contains personal or confidential information Perturbation methods are a set of mathematical techniques that can be used to protect data privacy by adding controlled noise or randomness to data while still allowing data analysis. These methods can be applied to various fields, including machine learning, statistics, and cryptography, to prevent attackers from identifying individuals or sensitive information. This review provides an overview of perturbation methods used to protect data privacy, including their advantages and limitations, and the importance of careful implementation to ensure accuracy and prevent bias.

Data privacy has become a critical concern in today's information age, as organizations collect and store vast amounts of data about individuals. While data analysis can provide valuable insights and improve decision-making, it also poses a risk to individuals' privacy. Perturbation methods offer a promising approach to protecting data privacy while still allowing data analysis. These methods involve adding controlled noise or randomness to data to preserve privacy. In this review, we will provide an overview of perturbation methods used to protect data privacy, including randomized response, differential privacy, secure multi-party computation (SMC), noise addition, and sampling and aggregation. We will also discuss the benefits and limitations of these methods and their potential applications in various fields. By understanding the various perturbation methods available for data privacy protection, researchers and practitioners can make informed decisions about how to protect sensitive information while still allowing data analysis.

This study organized into five sections. A literature review of the perturbation methods for data privacy in section 2. Section 3 presents the models and Section 4 covers the conclusion.

2. Literature Survey

A significant amount of research has been conducted on perturbation methods used to protect data privacy. The following literature survey highlights some of the key findings and contributions of various studies in this field:

1) Randomized Response: This technique adds randomness to the responses of individuals in a survey or questionnaire, making it difficult for an attacker to determine the true response [1,2]. Warner in 1965 first introduced to protect individual privacy in surveys [3]. Since then, it has been widely used in various fields, including healthcare, social sciences, and marketing. A study by Ghosh and Roth (2011) proposed a generalized randomized response method that provides better privacy guarantees than the original method [4,5].

2) Differential Privacy: It adds random noise to the data to prevent attackers from identifying individuals. This technique can be applied to a range of data analysis techniques, such as machine learning, statistics, and data mining. Dwork et al. (2006) first introduced protecting privacy [6]. Since then, it has become a popular method for protecting sensitive information. Wang et al. (2019) proposed a differential privacy algorithm for deep learning models that offers stronger privacy guarantees than existing methods [7]. The most common types of noise for differential privacy is the Laplace, exponential and Gaussian mechanism. They work by adding noise to the original data entry and can be applied to both real and categorical features. The Laplace strategy is a symmetric version of the exponential distribution, and it adds noise from a symmetric continuous distribution to the true answer according to equation 1 [8].

$$\mathrm{Lap}(x|b) = \frac{1}{2b}\exp\left(-\frac{|x|}{b}\right) \tag{1}$$

The exponential mechanism, on the other hand, selects and outputs an element $r \in R$ with probability proportional to equation 2.

$$\exp\left(\frac{\epsilon u(x,r)}{2\Delta u}\right) \tag{2}$$

where x is an input and u is a utility function with generalized sensitivity Δu.

3) Secure Multi-Party Computation (SMC): SMC has been widely used in privacy-preserving data analysis, where data from different sources is combined to perform a joint analysis. A study by Chaum et al. (1988) proposed a practical approach for secure computation of statistical functions using SMC [9]. Lindell and Pinkas (2000) proposed a practical SMC protocol that is widely used in various applications[10, 11,12].

4) Noise Addition: This technique involves adding a small amount of random noise to the data before releasing it for analysis. Some of the researchers proposed a noise addition-based approach for privacy-preserving principal component analysis that ensures data privacy while maintaining data utility [13, 14. 15, 16].

5) Sampling and Aggregation: Sampling involves selecting a subset of data to analyse, while aggregation involves combining data from multiple sources to perform an analysis. These techniques can be used to reduce the risk of sensitive information being disclosed while still allowing for accurate data analysis [17, 18]. The effectiveness of this method has been studied in various applications, including data mining and machine learning [2].

6) Privacy-Preserving Machine Learning: The privacy risks associated with machine learning algorithms and presents various perturbation methods to protect data privacy in machine learning. The authors discuss the advantages and limitations of each method and highlight their applications in machine learning [19, 20].

7) Privacy-Preserving Data Mining: Charu et al. provides an overview of various perturbation methods, including differential privacy, randomization, and secure multi-party computation [21].

In addition to these methods, other perturbation techniques have also been proposed, including data swapping, data masking, and k-anonymity. These techniques have been studied in various applications and have shown promising results for protecting data privacy.

3. Model and analysis

The perturbation method depends on the specific data analysis task and the desired level of privacy protection. Differential privacy provides a strong privacy guarantee, but may be computationally expensive and result in reduced data accuracy. Randomized response and noise addition offer a tuneable trade-off between privacy and accuracy, but may not provide strong privacy protection against more sophisticated attacks. Sampling and aggregation are computationally efficient and can be easily applied to large data sets, but may not provide strong privacy protection against more sophisticated attacks. It is important to carefully design and implement these methods to ensure that they do not compromise data accuracy or introduce bias into the results (in Figure 1).

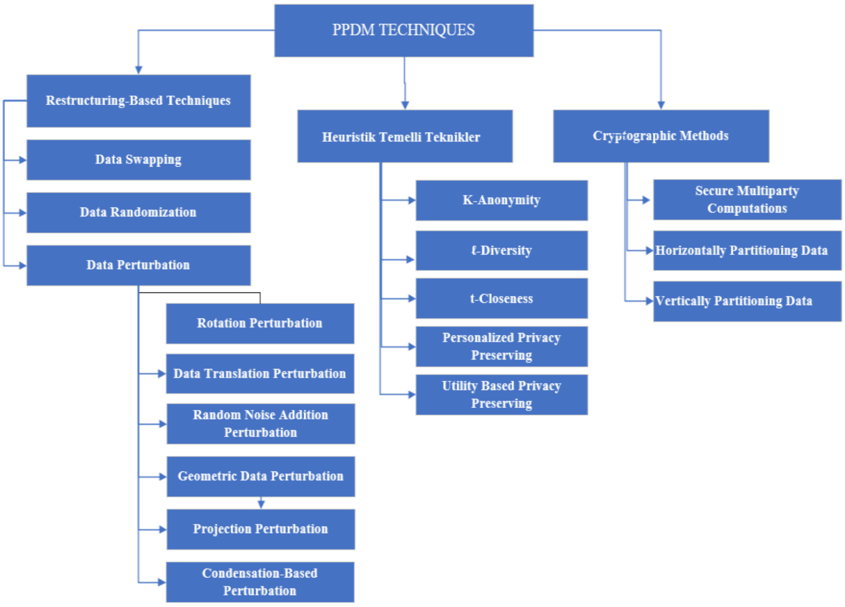
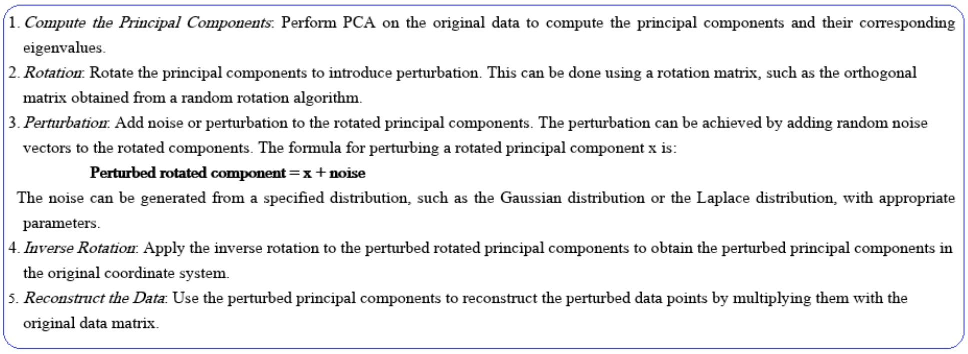

PPDM TECHNIQUES
Restructuring-Based Techniques
Data Swapping
Data Randomization
Data Perturbation
Rotation Perturbation
Data Translation Perturbation
Random Noise Addition Perturbation
Geometric Data Perturbation
Projection Perturbation
Condensation-Based Perturbation
Heuristik Temelli Teknikler
K-Anonymity
ℓ-Diversity
t-Closeness
Personalized Privacy Preserving
Utility Based Privacy Preserving
Cryptographic Methods
Secure Multiparty Computations
Horizontally Partitioning Data
Vertically Partitioning Data

Figure 1. Privacy Preserving Data Mining (PPDM) Techniques

In this section, data perturbation-rotation perturbation; principal component analysis(PCA); projection perturbation; geometric data perturbation; data swapping; data randomization; heuristic methods to protect data privacy; k-anonymity; k-anonymity l-diversity; k-anonymity l-closeness; personalized privacy preserving; utility based privacy preserving; cryptographic methods; secure multiparty computations; horizontally partitioning data; explanations of vertically partitioning data methods are given.

3.1. Data Perturbation-Rotation Perturbation

Rotation perturbation in Principal Component Analysis (PCA) is a technique used to add noise or perturbation to the principal components while preserving the overall structure of the data. It involves rotating the principal components and perturbing the rotated components. The specific formula for rotation perturbation in PCA depends on the perturbation method used in Figure 2.



1. *Compute the Principal Components*: Perform PCA on the original data to compute the principal components and their corresponding eigenvalues.
2. *Rotation*: Rotate the principal components to introduce perturbation. This can be done using a rotation matrix, such as the orthogonal matrix obtained from a random rotation algorithm.
3. *Perturbation*: Add noise or perturbation to the rotated principal components. The perturbation can be achieved by adding random noise vectors to the rotated components. The formula for perturbing a rotated principal component x is:

   Perturbed rotated component = x + noise

   The noise can be generated from a specified distribution, such as the Gaussian distribution or the Laplace distribution, with appropriate parameters.
4. *Inverse Rotation*: Apply the inverse rotation to the perturbed rotated principal components to obtain the perturbed principal components in the original coordinate system.
5. *Reconstruct the Data*: Use the perturbed principal components to reconstruct the perturbed data points by multiplying them with the original data matrix.

Figure 2: Data Perturbation Steps

The specific formulas for rotation perturbation in PCA may involve additional considerations depending on the chosen perturbation method and the desired level of perturbation. The goal is to introduce noise while preserving the overall structure and statistical properties of the data. It's important to select appropriate perturbation parameters and techniques to balance privacy protection and data utility in the perturbed data (in Figure 3).

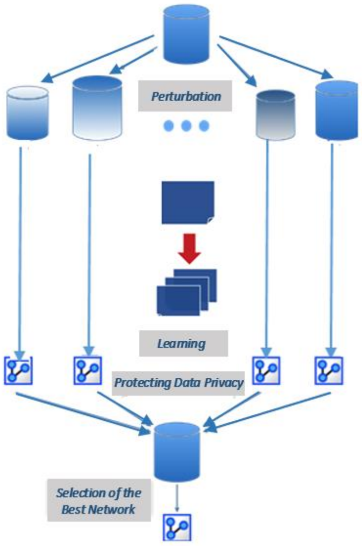


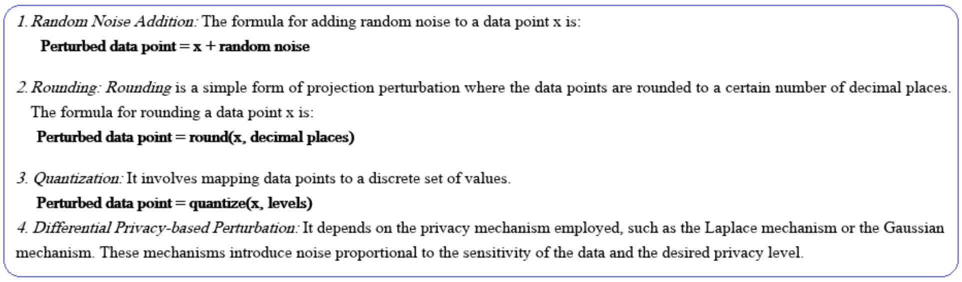
Figure 3: Perturbation Analysis Process

3.1.1 Rotation Perturbation Principal Component Analysis (PCA)

Projection perturbation is a technique used to add noise or perturbations to numerical data while preserving certain statistical properties. It involves projecting the data onto a lower-dimensional space and perturbing the projected values. The specific formula for projection perturbation depends on the perturbation method used.

3.1.2. Projection Perturbation

Data perturbation and projection perturbation are two techniques commonly used in data science and machine learning to protect data privacy. Data perturbation involves adding random noise to the data in order to protect the privacy of individual data points. The level of noise added can be controlled by a privacy budget, which balances privacy protection and data utility. Projection perturbation, on the other hand, involves projecting the data onto a lower-dimensional space while adding noise to the projection. This technique can help to remove identifying features of the data while preserving the overall structure and relationships between the data points. The choice of method depends on the specific application and privacy requirements. Additionally, the level of noise or dimensionality reduction used should be carefully chosen to balance privacy protection and data utility (in Figure 4).



1. *Random Noise Addition:* The formula for adding random noise to a data point x is:
Perturbed data point = x + random noise

2. *Rounding: Rounding* is a simple form of projection perturbation where the data points are rounded to a certain number of decimal places. The formula for rounding a data point x is:
Perturbed data point = round(x, decimal places)

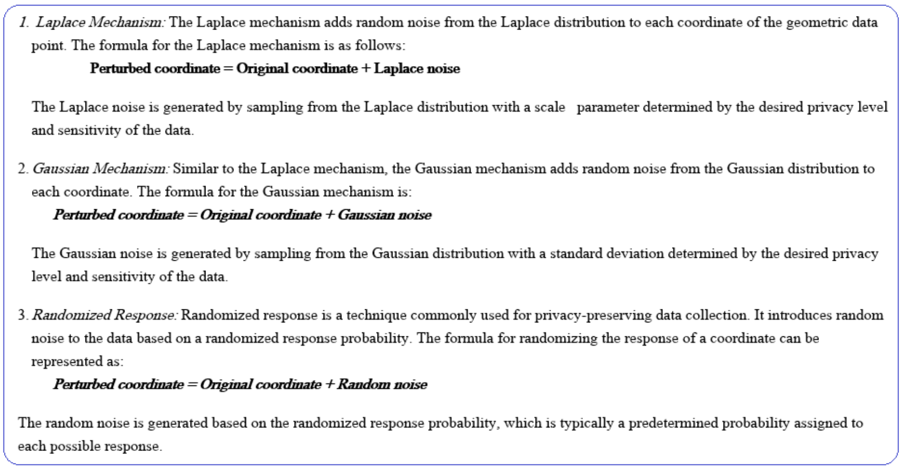
3. *Quantization:* It involves mapping data points to a discrete set of values.
Perturbed data point = quantize(x, levels)

4. *Differential Privacy-based Perturbation:* It depends on the privacy mechanism employed, such as the Laplace mechanism or the Gaussian mechanism. These mechanisms introduce noise proportional to the sensitivity of the data and the desired privacy level.

Figure 4: Projection Perturbation Steps

3.1.3. Geometric Data Perturbation

Geometric data perturbation is a technique used to add noise or perturbations to geometric data in order to protect privacy while preserving the general shape or structure of the data. The specific formula for geometric data perturbation may vary depending on the perturbation method used (in Fig.5).



1. *Laplace Mechanism:* The Laplace mechanism adds random noise from the Laplace distribution to each coordinate of the geometric data point. The formula for the Laplace mechanism is as follows:
Perturbed coordinate = Original coordinate + Laplace noise

The Laplace noise is generated by sampling from the Laplace distribution with a scale parameter determined by the desired privacy level and sensitivity of the data.

2. *Gaussian Mechanism:* Similar to the Laplace mechanism, the Gaussian mechanism adds random noise from the Gaussian distribution to each coordinate. The formula for the Gaussian mechanism is:
Perturbed coordinate = Original coordinate + Gaussian noise

The Gaussian noise is generated by sampling from the Gaussian distribution with a standard deviation determined by the desired privacy level and sensitivity of the data.

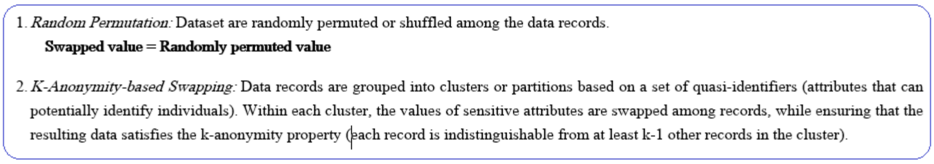
3. *Randomized Response:* Randomized response is a technique commonly used for privacy-preserving data collection. It introduces random noise to the data based on a randomized response probability. The formula for randomizing the response of a coordinate can be represented as:
Perturbed coordinate = Original coordinate + Random noise

The random noise is generated based on the randomized response probability, which is typically a predetermined probability assigned to each possible response.

Figure 5: Geometric Data Perturbation Steps

3.1.4. Data Swapping

Data swapping is a privacy-preserving technique used to protect sensitive information while preserving the statistical properties of the data. It involves swapping or exchanging values between data records in a way that maintains the overall data distribution but obscures the original relationships between individual records. The specific formula for data swapping depends on the swapping method used (in Figure 6).



1. *Random Permutation:* Dataset are randomly permuted or shuffled among the data records.

Swapped value = Randomly permuted value

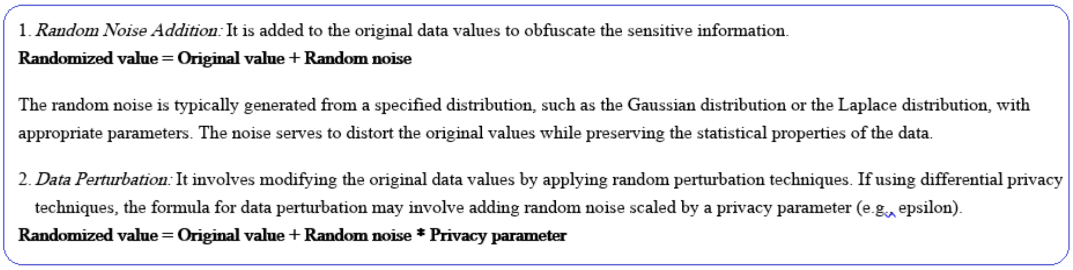
2. *K-Anonymity-based Swapping:* Data records are grouped into clusters or partitions based on a set of quasi-identifiers (attributes that can potentially identify individuals). Within each cluster, the values of sensitive attributes are swapped among records, while ensuring that the resulting data satisfies the k-anonymity property (each record is indistinguishable from at least k-1 other records in the cluster).

Figure 6: Data Swapping Steps

The formula for k-anonymity-based swapping involves selecting suitable records within a cluster and swapping the values of sensitive attributes. The exact implementation may vary depending on the specific algorithm used for k-anonymity.

3.1.5. Data Randomization

Data randomization is a technique used to protect data privacy by introducing random noise or perturbation to the original data values. The specific formula for data randomization depends on the randomization method used (in Figure 7).



1. *Random Noise Addition:* It is added to the original data values to obfuscate the sensitive information.

Randomized value = Original value + Random noise

The random noise is typically generated from a specified distribution, such as the Gaussian distribution or the Laplace distribution, with appropriate parameters. The noise serves to distort the original values while preserving the statistical properties of the data.

2. *Data Perturbation:* It involves modifying the original data values by applying random perturbation techniques. If using differential privacy techniques, the formula for data perturbation may involve adding random noise scaled by a privacy parameter (e.g. epsilon).

Randomized value = Original value + Random noise * Privacy parameter

Figure 7: Data Randomization Steps

The random noise is typically generated from a specific distribution, and the privacy parameter controls the level of privacy protection provided.

3.2 Heuristic Methods

Heuristic methods are commonly used in data science and machine learning to protect data privacy. These methods involve using general problem-solving techniques to develop strategies and rules for protecting sensitive data.

One example of a heuristic method for data privacy is k-anonymity, which is a technique used to ensure that each record in a dataset is indistinguishable from at least k-1 other records in the dataset. This involves grouping similar records together and removing any identifying information that could be used to link a record to a specific individual.

Another example is l-diversity, which is a technique used to ensure that each group of records with a given sensitive attribute value has at least l different values for another attribute. This helps to prevent attackers from linking sensitive attributes to specific individuals in the dataset. Other heuristic methods for data privacy include t-closeness, differential privacy, and machine learning-based techniques such as generative adversarial networks (GANs) and variational autoencoders (VAEs). While heuristic methods can be effective for protecting data privacy, it is important to carefully evaluate their effectiveness and to select appropriate methods and parameters based on the specific needs of the analysis and the privacy risks associated with the data.

3.2.1. k-Anonymity

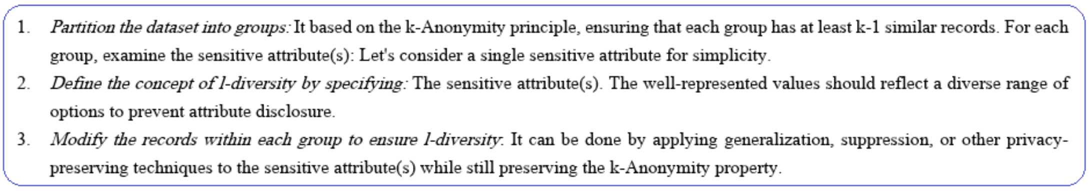
k-Anonymity is a privacy-preserving technique that aims to protect individual identities in a dataset by ensuring that each record in the dataset is indistinguishable from at least k-1 other records with respect to certain identifying attributes. The k-Anonymity principle helps to prevent the re-identification of individuals by reducing the uniqueness of their identifying information. The basic idea behind k-Anonymity is to generalize or suppress the values of attributes in a way that groups of records become indistinguishable while maintaining the overall statistical properties of the data. The specific formula for achieving k-Anonymity depends on the chosen generalization or suppression method.

It's important to note that achieving k-Anonymity requires careful consideration of the chosen attributes, the level of generalization or suppression, and the desired level of privacy protection. Additionally, the effectiveness of k-Anonymity depends on the quality of the generalization or suppression techniques applied and the size of the anonymized groups. Striking a balance between privacy protection and data utility is crucial in implementing k-Anonymity to ensure both privacy preservation and meaningful analysis of the anonymized data.

3.2.1.1. k-Anonymity l-Diversity

k-Anonymity aims to protect individual identities by ensuring that each record in a dataset is indistinguishable from at least k-1 other records. However, k-Anonymity alone may not be sufficient to prevent attribute disclosure. This is where l-diversity comes into play as an enhancement to k-Anonymity. l-diversity ensures that each group of indistinguishable records (based on k-Anonymity) contains at least l well-represented values for sensitive attributes.

The specific formula for achieving l-diversity depends on the chosen method and the definition of well-represented values (in Figure 8).



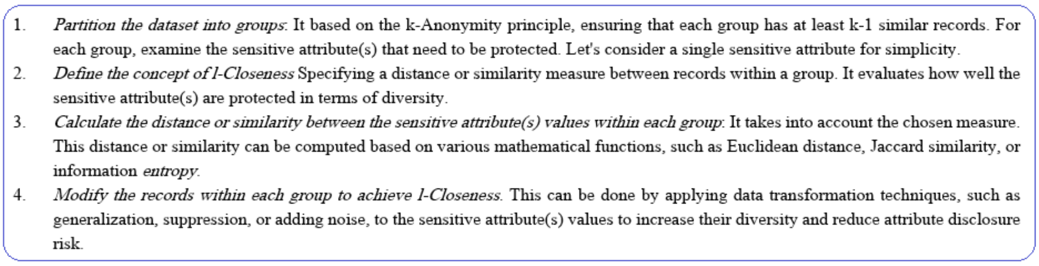
1. *Partition the dataset into groups:* It based on the k-Anonymity principle, ensuring that each group has at least k-1 similar records. For each group, examine the sensitive attribute(s): Let's consider a single sensitive attribute for simplicity.
2. *Define the concept of l-diversity by specifying:* The sensitive attribute(s). The well-represented values should reflect a diverse range of options to prevent attribute disclosure.
3. *Modify the records within each group to ensure l-diversity:* It can be done by applying generalization, suppression, or other privacy-preserving techniques to the sensitive attribute(s) while still preserving the k-Anonymity property.

Figure 8: k-Anonymity l-Diversity Steps

The key idea behind l-diversity is to ensure that within each group, the sensitive attribute(s) have a sufficient number of distinct, well-represented values to prevent attribute disclosure even if the group is still indistinguishable based on k-Anonymity.

3.2.1.2. k-Anonymity l-Closeness

k-Anonymity and l-Closeness are two complementary privacy protection techniques used to safeguard sensitive information in datasets. While k-Anonymity focuses on hiding the identity of individuals, l-Closeness aims to address attribute disclosure by ensuring that sensitive attributes in a group of records are sufficiently diverse (in Fig. 9).



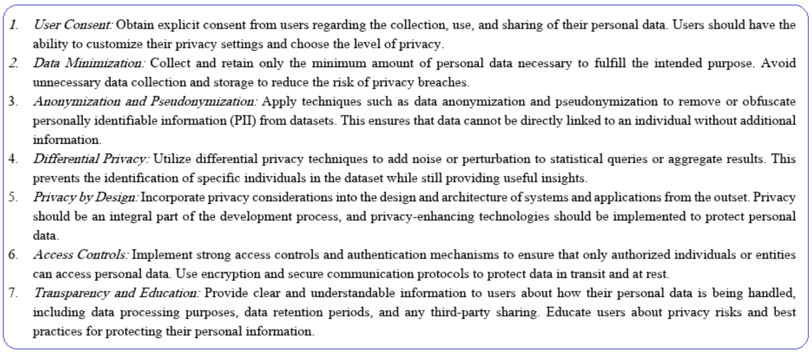
1. *Partition the dataset into groups:* It based on the k-Anonymity principle, ensuring that each group has at least k-1 similar records. For each group, examine the sensitive attribute(s) that need to be protected. Let's consider a single sensitive attribute for simplicity.
2. *Define the concept of l-Closeness* Specifying a distance or similarity measure between records within a group. It evaluates how well the sensitive attribute(s) are protected in terms of diversity.
3. *Calculate the distance or similarity between the sensitive attribute(s) values within each group:* It takes into account the chosen measure. This distance or similarity can be computed based on various mathematical functions, such as Euclidean distance, Jaccard similarity, or information *entropy*.
4. *Modify the records within each group to achieve l-Closeness.* This can be done by applying data transformation techniques, such as generalization, suppression, or adding noise, to the sensitive attribute(s) values to increase their diversity and reduce attribute disclosure risk.

Figure 9: K-Anonymity l-Closeness Steps

The exact formula for achieving l-Closeness within a group depends on the specific distance or similarity measure chosen and the selected data transformation technique. The goal is to ensure that the sensitive attribute(s) values within a group exhibit a diversity that satisfies the l-Closeness requirement. By combining k-Anonymity and l-Closeness, privacy protection can be strengthened as k-Anonymity ensures the indistinguishability of records, while l-Closeness addresses the risk of attribute disclosure by enforcing diversity within each group.

3.2.2. Personalized Privcy Preserving

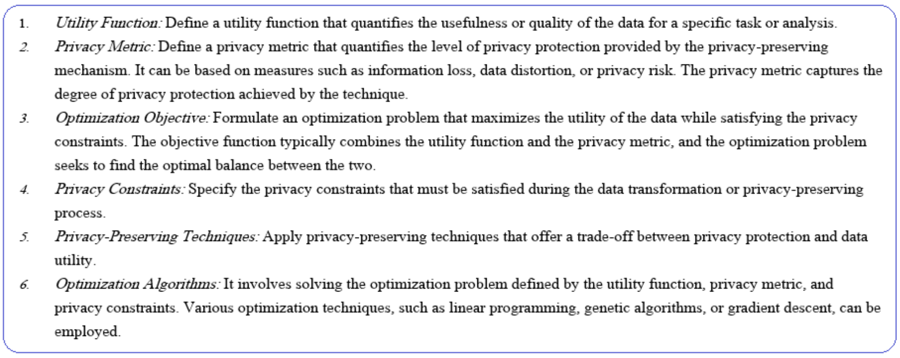
Personalized Privacy Preserving (P3) is a heuristic method used to protect data privacy that focuses on preserving the privacy of individuals rather than the privacy of the overall dataset. The goal of P3 is to allow data analysts to extract useful information from a dataset while minimizing the risk of disclosing sensitive information about individuals. To implement P3, each individual in the dataset is assigned a personalized privacy parameter that determines the level of privacy protection they receive. This parameter is based on the individual's risk of identity disclosure, which is calculated based on the uniqueness of their attributes in the dataset. Individuals with highly unique attributes receive higher levels of privacy protection, while those with less unique attributes receive lower levels of protection. Personalized Privacy Preserving refers to the concept of tailoring privacy protection mechanisms to the individual preferences and requirements of data subjects. It aims to provide users with control over their personal information while still allowing them to benefit from data analysis and services (in Figure 10).



1. *User Consent:* Obtain explicit consent from users regarding the collection, use, and sharing of their personal data. Users should have the ability to customize their privacy settings and choose the level of privacy.
2. *Data Minimization:* Collect and retain only the minimum amount of personal data necessary to fulfill the intended purpose. Avoid unnecessary data collection and storage to reduce the risk of privacy breaches.
3. *Anonymization and Pseudonymization:* Apply techniques such as data anonymization and pseudonymization to remove or obfuscate personally identifiable information (PII) from datasets. This ensures that data cannot be directly linked to an individual without additional information.
4. *Differential Privacy:* Utilize differential privacy techniques to add noise or perturbation to statistical queries or aggregate results. This prevents the identification of specific individuals in the dataset while still providing useful insights.
5. *Privacy by Design:* Incorporate privacy considerations into the design and architecture of systems and applications from the outset. Privacy should be an integral part of the development process, and privacy-enhancing technologies should be implemented to protect personal data.
6. *Access Controls:* Implement strong access controls and authentication mechanisms to ensure that only authorized individuals or entities can access personal data. Use encryption and secure communication protocols to protect data in transit and at rest.
7. *Transparency and Education:* Provide clear and understandable information to users about how their personal data is being handled, including data processing purposes, data retention periods, and any third-party sharing. Educate users about privacy risks and best practices for protecting their personal information.

Figure 10: Personalized Privacy Preserving

3.2.3. Utility Based Privacy Preserving



1. *Utility Function:* Define a utility function that quantifies the usefulness or quality of the data for a specific task or analysis.
2. *Privacy Metric:* Define a privacy metric that quantifies the level of privacy protection provided by the privacy-preserving mechanism. It can be based on measures such as information loss, data distortion, or privacy risk. The privacy metric captures the degree of privacy protection achieved by the technique.
3. *Optimization Objective:* Formulate an optimization problem that maximizes the utility of the data while satisfying the privacy constraints. The objective function typically combines the utility function and the privacy metric, and the optimization problem seeks to find the optimal balance between the two.
4. *Privacy Constraints:* Specify the privacy constraints that must be satisfied during the data transformation or privacy-preserving process.
5. *Privacy-Preserving Techniques:* Apply privacy-preserving techniques that offer a trade-off between privacy protection and data utility.
6. *Optimization Algorithms:* It involves solving the optimization problem defined by the utility function, privacy metric, and privacy constraints. Various optimization techniques, such as linear programming, genetic algorithms, or gradient descent, can be employed.

Figure 11: Utility Based Privacy Preserving Steps

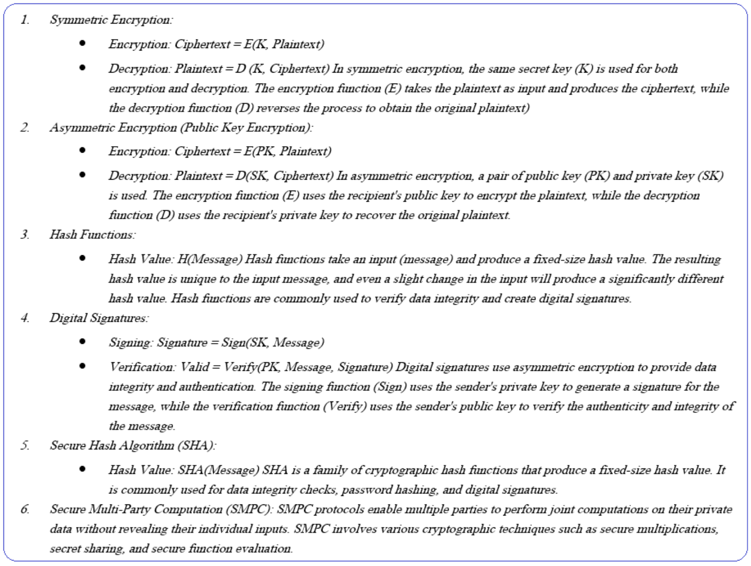
Utility-based privacy preserving is a heuristic method used to protect data privacy by balancing privacy protections with the utility (or usefulness) of the data. This approach recognizes that complete privacy protection may not always be feasible or desirable, particularly in situations where data is needed for research or other purposes. To implement utility-based privacy preserving, data is first assessed to determine the level of privacy protection required based on the sensitivity of the data and the risks associated with disclosure. Then, data is processed to ensure that privacy protections are applied to the appropriate data elements, while minimizing the impact on data utility. In utility-based privacy preserving, the focus is on optimizing the trade-off between data privacy and data utility (in Fig. 11).

3.3. Cryptographic Methods

Cryptographic methods are a class of heuristic methods used to protect data privacy that involve the use of encryption and decryption techniques to secure sensitive data. These methods use mathematical algorithms to encode data in such a way that it can only be accessed by authorized individuals or systems with the appropriate decryption keys. They can be broadly divided into two categories: symmetric key cryptography and public key cryptography. One common method used for data privacy protection is symmetric-key encryption, where the same key is used for both encryption and decryption. In this method, the data is encrypted using a secret key that is shared between the data owner and the authorized recipient. The data is then transmitted securely over a network or stored on a device, and can only be accessed by those with the appropriate decryption key. Another method used for data privacy protection is public-key encryption, where two different keys are used for encryption and decryption. In this method, a public key is used for encrypting data, while a private key is used for decryption. (In Figure 12).

3.3.1. Secure Multiparty Computations (MPC)

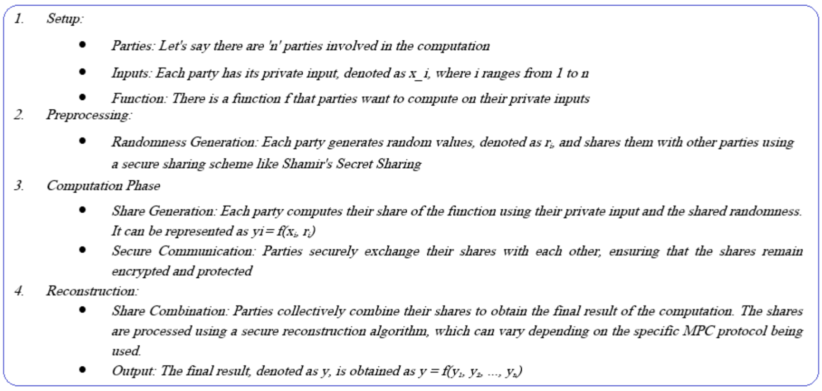
In an MPC protocol, each party holds their own private data and wants to compute a function over the combined data without sharing their data with other parties. To achieve this, the parties interact with each other to perform the computation in a way that preserves privacy.



1. *Symmetric Encryption:*
   - *Encryption: Ciphertext = E(K, Plaintext)*
   - *Decryption: Plaintext = D (K, Ciphertext) In symmetric encryption, the same secret key (K) is used for both encryption and decryption. The encryption function (E) takes the plaintext as input and produces the ciphertext, while the decryption function (D) reverses the process to obtain the original plaintext)*
2. *Asymmetric Encryption (Public Key Encryption):*
   - *Encryption: Ciphertext = E(PK, Plaintext)*
   - *Decryption: Plaintext = D(SK, Ciphertext) In asymmetric encryption, a pair of public key (PK) and private key (SK) is used. The encryption function (E) uses the recipient's public key to encrypt the plaintext, while the decryption function (D) uses the recipient's private key to recover the original plaintext.*
3. *Hash Functions:*
   - *Hash Value: H(Message) Hash functions take an input (message) and produce a fixed-size hash value. The resulting hash value is unique to the input message, and even a slight change in the input will produce a significantly different hash value. Hash functions are commonly used to verify data integrity and create digital signatures.*
4. *Digital Signatures:*
   - *Signing: Signature = Sign(SK, Message)*
   - *Verification: Valid = Verify(PK, Message, Signature) Digital signatures use asymmetric encryption to provide data integrity and authentication. The signing function (Sign) uses the sender's private key to generate a signature for the message, while the verification function (Verify) uses the sender's public key to verify the authenticity and integrity of the message.*
5. *Secure Hash Algorithm (SHA):*
   - *Hash Value: SHA(Message) SHA is a family of cryptographic hash functions that produce a fixed-size hash value. It is commonly used for data integrity checks, password hashing, and digital signatures.*
6. *Secure Multi-Party Computation (SMPC): SMPC protocols enable multiple parties to perform joint computations on their private data without revealing their individual inputs. SMPC involves various cryptographic techniques such as secure multiplications, secret sharing, and secure function evaluation.*

Figure 12: Cryptographic method steps

The key idea behind MPC is that even though each party contributes their private input and partial computation, no party can determine the inputs of other parties or the intermediate results. The protocol ensures privacy, confidentiality, and integrity of the inputs throughout the computation (in Figure 13).



1. *Setup:*
    - *Parties: Let's say there are 'n' parties involved in the computation*
    - *Inputs: Each party has its private input, denoted as x_i, where i ranges from 1 to n*
    - *Function: There is a function f that parties want to compute on their private inputs*
2. *Preprocessing:*
    - *Randomness Generation: Each party generates random values, denoted as r_i, and shares them with other parties using a secure sharing scheme like Shamir's Secret Sharing*
3. *Computation Phase*
    - *Share Generation: Each party computes their share of the function using their private input and the shared randomness. It can be represented as yi = $f(x_i, r_i)$*
    - *Secure Communication: Parties securely exchange their shares with each other, ensuring that the shares remain encrypted and protected*
4. *Reconstruction:*
    - *Share Combination: Parties collectively combine their shares to obtain the final result of the computation. The shares are processed using a secure reconstruction algorithm, which can vary depending on the specific MPC protocol being used.*
    - *Output: The final result, denoted as y, is obtained as $y = f(y_1, y_2, \ldots, y_n)$*

Figure 13: Secure Multiparty Computations (MPC) Steps

3.3.2. Horizontally Partitioning Data

Horizontally partitioning data is a technique used in data privacy to distribute and store different attributes or subsets of data across multiple data sources while preserving privacy.

3.3.3. Vertically Partitioning Data

Vertically partitioning data is a technique used in data privacy to split a dataset vertically into multiple subsets based on different attributes or columns. Each subset contains a subset of the original attributes, preserving the privacy of certain sensitive attributes. The specific formulas for vertically partitioning data depend on the privacy-preserving technique being used and the specific attributes and privacy requirements of the dataset. Various algorithms and methodologies can be employed to determine the optimal partitioning strategy and achieve the desired privacy guarantees.

4. Conclusion

In conclusion, there are various perturbation methods that can be used to protect data privacy, each with its own strengths and weaknesses. Randomized response and noise addition are simple and effective perturbation methods, but they can be vulnerable to certain types of attacks and may require careful tuning of the noise level to balance privacy protection and data utility. Differential privacy provides a strong privacy guarantee, but can be computationally expensive. Secure multi-party computation provides strong privacy protection without requiring data to be perturbed or modified, but can also be computationally expensive.

The choice of perturbation method will depend on the specific application and the trade-off between privacy protection and data utility. Researchers continue to explore and develop new perturbation methods and optimization techniques to improve the privacy and utility of data analysis in various settings.

References

[1] Newman M. E. J. The structure and function of complex networks, SIAM Rev., 45 (2) (2003), pp. 167-256

[2] Kargupta Hillol & Datta Souptik & Wang Q. & Sivakumar Krishnamoorthy. (2003). On the privacy preserving properties of random data perturbation techniques. Proceedings - IEEE International Conference on Data Mining, ICDM. 99- 106. 10.1109/ICDM. 2003.1250908.

[3] Warner S.L. Randomized response: a survey technique for eliminating evasive answer bias. Journal of the American Statistical Association 60 (1965) (309), 63-69

[4] Ghosh A., Roughgarden T. and Sundararajan M. "Universally utility-maximizing privacy mechanisms", SIAM J. Comput., vol. 41 (2012), no. 6, pp. 1673-1693.

[5] Xu L., C. Jiang J. Wang J. Yuan and Y. Ren. Information Security in Big Data: Privacy and Data Mining. IEEE Access, Vol. 2, (2014), pp. 1149–1176.

[6] Dwork, C.; McSherry, F.; Nissim, K.; Smith, A. Calibrating noise to sensitivity in private data analysis. In Theory of Cryptography Conference; Springer: Berlin/Heidelberg, Germany, (2006); pp. 265–284

[7] Wang, T.; Zheng, Z.; Rehmani, M.H.; Yao, S.; Huo, Z. Privacy Preservation in Big Data from the Communication Perspective—A Survey. IEEE Commun. Surv. Tutor. (2019), 21, 753–778.

[8] Liu C, Chen S, Zhou S, et al (2019) A novel privacy preserving method for data publication. Inf Sci 501:421–435. https://doi.org/10.1016/j.ins.2019.06.022

[9] Chaum D., Crepeau C., Damgard I., (1988). Multiparty Unconditionally Secure Protocols. Proc. 20th Annual ACM Symp. On Theory of Computing, p.11–19. [doi:10.1145/62212.62214]

[10] Lindell Y. and Pinkas B., Privacy preserving data mining. In Advances in Cryptology – CRYPTO '00, volume 1880 of Lecture Notes in Computer Science, pages 36–54. Springer-Verlag, 2000.

[11] Mohassel P. and M. Franklin, Efficiency Tradeoffs for Malicious Two-Party Computation, in Public Key Cryptography - PKC (2006), p. 458-473.

[12] Yang X., Feng Y., Fang W., Shao J., An Accuracy-Lossless Perturbation Method for Defending Privacy Attacks in Federated Learning , Network and Distributed Systems Security (NDSS) Symposium 2021 21-24 February 2021 ISBN 1-891562-66-5 https://dx.doi.org/10.14722/ndss.2021.23xxx www.ndss-symposium.org

[13] Zhang X. Impacts of different perturbation methods on multiscale interactions between multisource perturbations for convection-permitting ensemble forecasting during SCMREX. Q J R Meteorol Soc, 147(2021), 741, 3899– 3921. Available from: https://doi.org/10.1002/qj.4160

[14] Shokri R., Theodorakopoulos G., Le Boudec J.Y., et al. Quantifying location privacy, 32nd IEEE Symposium on Security and Privacy (2011), pp. 247-262

[15] Li G, Xue R., A new privacy-preserving data mining method using non-negative matrix factorization and singular value decomposition. Wireless Personal Commun 102 (2018) (2), pp. 1799–1808.

[16] Yang D., Qu B., Cudré-Mauroux P., Privacy-Preserving Social Media Data Publishing for Personalized Ranking-Based Recommendatio, IEEE Trans. Knowl. Data Eng. (2018)

[17] Xiao X.; Bender G.; Hay M.; Gehrke J. iReduct: Differential privacy with reduced relative errors. In Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, Athens, Greece, 12–16 (June 2011), pp. 229–240.

[18] Kairouz Peter & Oh S. & Viswanath P. (2016). Extremal mechanisms for local differential privacy. 17.

[19] Chhinkaniwala, H. & Garg, S." Tuple -Value Based Multiplicative Data Perturbtion Approach to preserve privacy in data stream mining", IJDKP, Vol3, (May 2013), No.3.

[20] Li C., Palanisamy B., Reversible spatio-temporal perturbation for protecting location privacy, Computer Communications, Volume 135, (2019), pp.16-27, ISSN 0140-3664, https://doi.org/10.1016/j.comcom.2018.12.003.

[21] Shokri R. Quantifying and Protecting Location Privacy, THÈSE NO 5622 (2013) École Polytechnique Fédérale De Lausanne Présentée le 8 mars 2013