

Data privacy protection methods

104369: Privacitat de Dades i Seguretat

G. Navarro-Arribas

UAB -- DEIC

v. 20260928

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

Summary

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

- Rank Swapping

- Microaggregation

- Noise

- Generalization

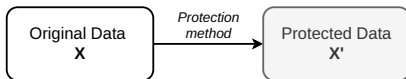
- Other methods

- Synthetic Data Generators

Summary

Protection methods (masking methods)

General case:



- We will see methods for different scenarios and models mixed up.
- Most of them can be used or adapted for interactive/non-interactive scenarios.
- Here we focus on the offline/non-interactive scenario.

Protection methods classification

■ **Perturbative:**

- The original data X is distorted.
- Protected data X' contain some erroneous information.
- E.g. rank swapping, microaggregation, noise addition, PRAM.

■ **Non-perturbative:**

- Replace original values in X by others which are less specific (or remove them).
- Protected data X' do not contain erroneous information.
- E.g. generalization (recoding), suppression, top/bottom coding, sampling.

■ **Synthetic data generators:**

- New artificial data is generated to substitute the original data.

Protection methods vs. privacy models

Protection method

- An algorithm: *how* X is transformed into X' .
- E.g. microaggregation, noise, generalization.
- No guarantee by itself: the privacy achieved depends on the parameters and has to be evaluated.

Privacy model

- A definition: *what* property the protected data (or the release mechanism) must satisfy.
- E.g. k -anonymity, ℓ -diversity, differential privacy.
- A formal guarantee, independent of how it is achieved.

- Methods are the *tools* used to satisfy a model:
 - Microaggregation or generalization $\Rightarrow k$ -anonymity.
 - Laplace noise calibrated to the query sensitivity $\Rightarrow$ differential privacy.

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

- Rank Swapping

- Microaggregation

- Noise

- Generalization

- Other methods

- Synthetic Data Generators

Summary

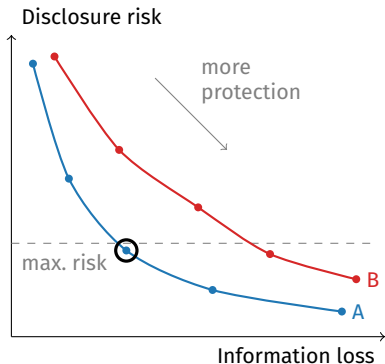
Evaluation of protection methods

How do we know how good a given method is?

Privacy vs. Utility

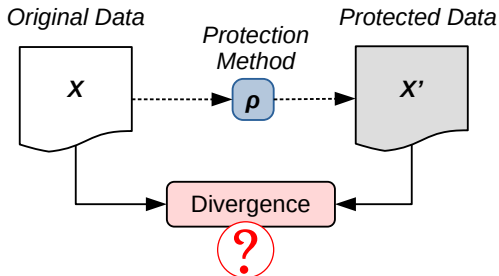
- **Privacy:** level of privacy or anonymity achieved by the protection.
 - Privacy model: does it satisfy some privacy model? (k -anonymity, differential privacy)
 - Reidentification: simulate an attack. How many users can the attacker re-identify? A possible approach is using *record linkage* (example later).
- **Utility or Information loss:** Usually greater privacy implies greater distortion or generalization of the original data. We need to measure how information is deteriorated.

Risk--utility (R-U) map



- Each point: a method with one parameter setting ($k, p, \dots$).
- More protection moves along the curve: less risk, more loss.
- Choose the setting with the least loss among those below the acceptable risk (circled).
- Here A dominates B: for the same risk it loses less information.

Information Loss



- **Generic** information loss
- **Specific** information loss

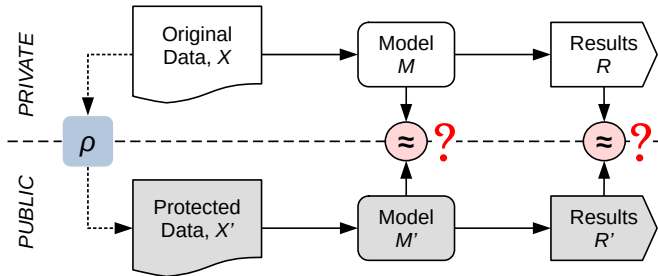
Generic Information Loss

Generic information loss: no specific use of the data

- Measure **divergence** $div(X, X')$, where *div* can be any sensible error measure:
 - Error (difference) on concrete statistics: mean, variance, covariance, correlation coefficient, ...
 - MSE (Mean Squared Error), MAE (Mean Absolute Error), MRE (Mean Relative Error), ...
 - ...

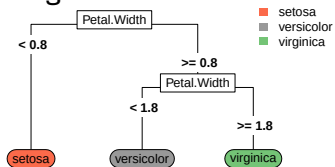
Specific information loss

- Evaluation of the information lost for a specific task/purpose.
- Commonly a machine learning model:

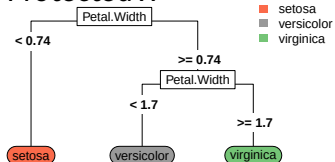


Example specific information loss i

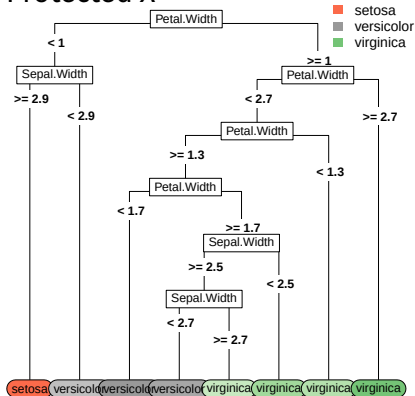
Original X



Protected X'



Protected X''



Example specific information loss ii

<i>Petal.Width</i>	<i>Sepal.Width</i>	<i>M</i>	<i>M'</i>	<i>M''</i>
0.5	3.0	<i>setosa</i>	<i>setosa</i>	<i>setosa</i>
0.2	3.5	<i>setosa</i>	<i>setosa</i>	<i>setosa</i>
1.3	2.7	<i>versicolor</i>	<i>versicolor</i>	<i>versicolor</i>
1.8	3.2	<i>virginica</i>	<i>virginica</i>	<i>virginica</i>
2.3	3.0	<i>virginica</i>	<i>virginica</i>	<i>virginica</i>
0.5	2.1	<i>setosa</i>	<i>setosa</i>	<i>versicolor</i>
2.1	2.6	<i>virginica</i>	<i>virginica</i>	<i>versicolor</i>
1.7	3.2	<i>versicolor</i>	<i>virginica</i>	<i>virginica</i>

- Agreement with *M*: *M'* in 7 of 8 records, *M''* in 5 of 8.
- Protection can also change the *complexity* of the model: compare the depth of the tree for *X''* with the one for *X*.

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

Rank Swapping

Microaggregation

Noise

Generalization

Other methods

Synthetic Data Generators

Summary

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

Rank Swapping

Microaggregation

Noise

Generalization

Other methods

Synthetic Data Generators

Summary

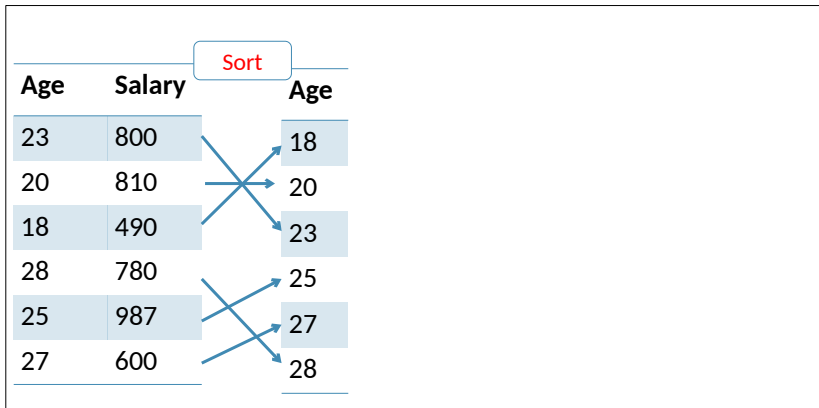
Rank Swapping

- Given a single attribute (variable) $V = (a_1, \dots, a_n)$ and a parameter p (a percentage of n):
 - Sort the values of V in ascending order.
 - From the lowest rank upwards, each value at rank i not yet swapped is swapped with a randomly chosen unswapped value at rank l , with $i < l \leq i + \lfloor p n / 100 \rfloor$.
 - Undo the sorting step.
- Repeat independently for each attribute.
- Values only move between records: each attribute keeps exactly its original values (same mean, variance, distribution).
- Relationships between attributes (e.g. correlations) are distorted.
- Larger $p \Rightarrow$ values travel further $\Rightarrow$ more privacy, less utility [14].

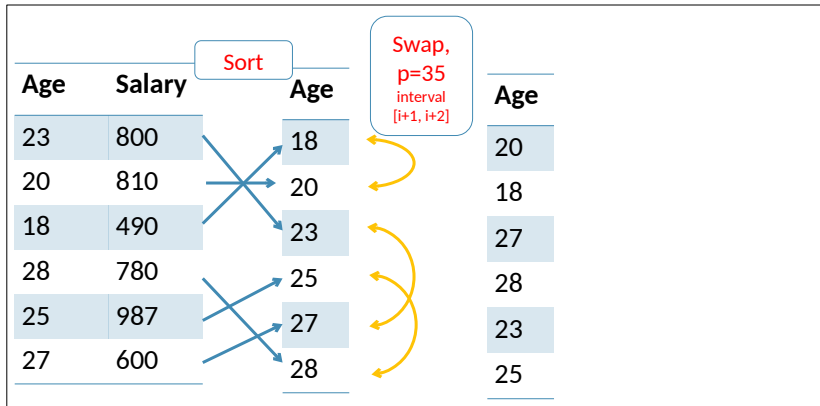
Rank Swapping example

Age	Salary
23	800
20	810
18	490
28	780
25	987
27	600

Rank Swapping example

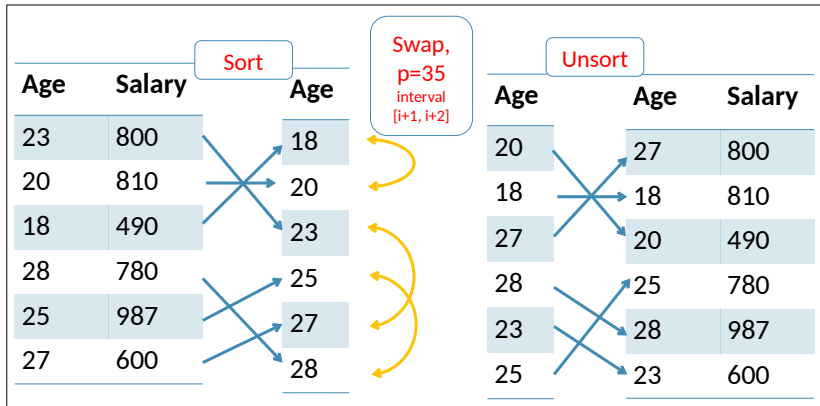


Rank Swapping example



$n = 6, p = 35: \lfloor 35 \cdot 6/100 \rfloor = 2 \Rightarrow \text{partner rank in } [i + 1, i + 2].$

Rank Swapping example



$n = 6, p = 35: \lfloor 35 \cdot 6/100 \rfloor = 2 \Rightarrow \text{partner rank in } [i + 1, i + 2].$

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

Rank Swapping

Microaggregation

Noise

Generalization

Other methods

Synthetic Data Generators

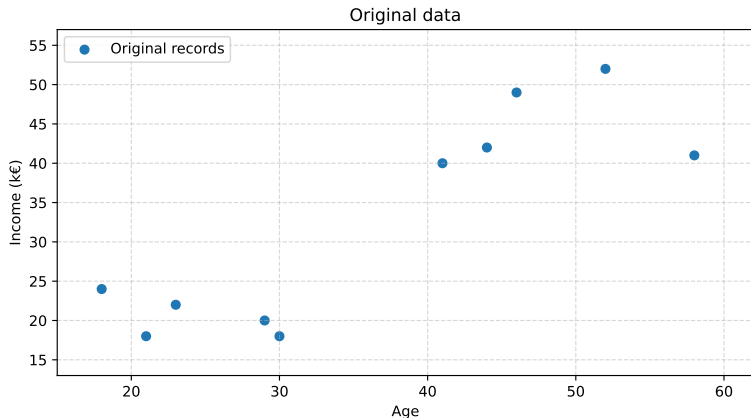
Summary

Microaggregation

- Build small microclusters and replace each original record by its cluster representative.
- Usual method:
 - **Partition:** group **similar** records into clusters of at least k records (and at most $2k - 1$)
 - **Aggregation:** compute the cluster **representative** (centroid)
 - **Replacement:** each record is replaced by its cluster representative.

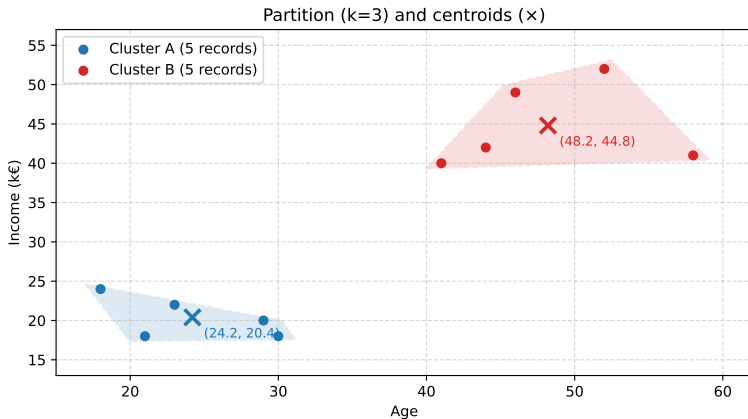
Microaggregation example

Quasi-identifiers: age, income.



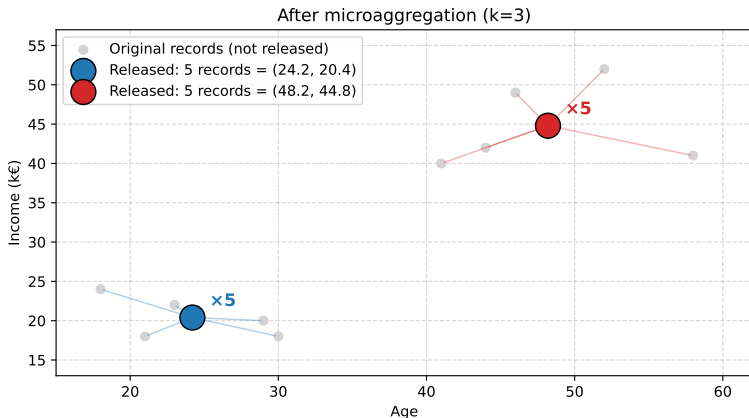
Microaggregation example

Quasi-identifiers: age, income.



Microaggregation example

Quasi-identifiers: age, income.



Each released (age, income) pair appears 5 times $\geq k$.

Microaggregation example

Quasi-identifier: age (univariate microaggregation).

Age	Salary
23	800
20	810
18	490
28	780
25	987
27	600

Microaggregation example

Quasi-identifier: age (univariate microaggregation).

Age	Salary	Partition k=2	Age
23	800		23
20	810		20
18	490		18
28	780		28
25	987		25
27	600		27

Microaggregation example

Quasi-identifier: age (univariate microaggregation).

Age	Salary	Partition k=2	Aggregation, replacement
23	800	23	24
20	810	20	19
18	490	18	19
28	780	28	27.5
25	987	25	24
27	600	27	27.5

Microaggregation example

Quasi-identifier: age (univariate microaggregation).

		Partition k=2	Aggregation, replacement			
Age	Salary		Age	Age	Age	Salary
23	800	23	24	24	24	800
20	810	20	19	19	19	810
18	490	18	19	19	19	490
28	780	28	27.5	27.5	27.5	780
25	987	25	24	24	24	987
27	600	27	27.5	27.5	27.5	600

Microaggregation as an optimization problem

$$\min_{\chi, p} \text{SSE}(\chi, p; X, c, d) = \sum_{i=1}^c \sum_{x \in X} \chi_i(x) (d(x, p_i))^2$$

Subject to:

$$\sum_{i=1}^c \chi_i(x) = 1 \quad \text{for all } x \in X,$$

$$2k > \sum_{x \in X} \chi_i(x) \geq k \quad \text{for all } i = 1, \dots, c,$$

$$\chi_i(x) \in \{0, 1\} \quad \text{for all } x \in X, \forall i = 1, \dots, c.$$

Where:

- Dataset $X = \{x_1, \dots, x_n\}$, with $x \in \mathbb{R}^m$ (m attributes).
- c clusters with representatives $p_1, \dots, p_c$.
- Distance function $d(\cdot, \cdot)$ (e.g., Euclidean). With the Euclidean distance, the optimal p_i is the centroid (mean) of cluster i .
- Assignment $\chi_i(x) \in \{0, 1\}$: $\chi_i(x) = 1$ if x is in cluster i .

Some notes on microaggregation

- **Univariate** (one attribute at a time) or **multivariate** (several attributes jointly).
- Optimal univariate microaggregation: polynomial time [7].
- Optimal multivariate microaggregation is NP-hard [16]!
We use heuristic methods:
 - MDAV (maximum distance to average vector) [1],
V-MDAV [21], ...
 - Mondrian [11] (median cuts, see later).
- Gives ***k*-anonymity** only if *all* quasi-identifiers are microaggregated jointly (not attribute by attribute).
- Does not prevent *attribute disclosure*: if the records of a cluster share the same sensitive value, it is revealed ($\rightarrow$ ℓ -diversity [12]).

MDAV heuristic

While at least $3k$ records remain:

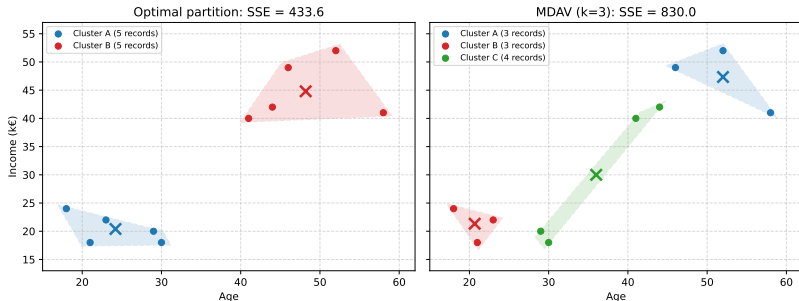
1. Compute the centroid $\bar{x}$ of the remaining records.
2. x_r : record farthest from $\bar{x}$. x_s : record farthest from x_r .
3. Form a cluster with x_r and its $k - 1$ nearest records, and another one with x_s and its $k - 1$ nearest records. Remove them.

Then:

- If between $2k$ and $3k - 1$ records remain: one cluster with the record farthest from $\bar{x}$ and its $k - 1$ nearest records, and another one with the rest.
 - If fewer than $2k$ remain: they form the last cluster.
- Simple and fast ($O(n^2)$), but all clusters have size k (except the last one), even if the data suggest otherwise.

MDAV vs. optimal partition

Same data as before, $k = 3$:



- MDAV forms groups of exactly 3: the last group of 4 straddles the two natural groups, and the SSE almost doubles.

Mondrian partitioning

Quasi-identifiers with totally ordered domains $D_1, \dots, D_m$; X is a multiset of points in $D_1 \times \dots \times D_m$.

- **Strict multidimensional partition:** a set of disjoint axis-aligned boxes $R_1, \dots, R_c$ that cover $D_1 \times \dots \times D_m$.
- It gives k -anonymity if every box contains either 0 or at least k records: $|X \cap R_i| \geq k$ for every non-empty R_i .
- A cut of R on attribute j at value v , $R^- = \{x \in R : x_j \leq v\}$ and $R^+ = \{x \in R : x_j > v\}$, is **allowable** if both $|X \cap R^-| \geq k$ and $|X \cap R^+| \geq k$.

Greedy algorithm [11] (start with $R = D_1 \times \dots \times D_m$):

1. Choose the attribute j with the widest (normalized) range in R .
 2. Cut at the median of the values x_j , $x \in X \cap R$.
 3. If the cut is allowable, recurse on R^- and R^+ ; otherwise try another attribute. If no cut is allowable, R is a final box.
- $O(n \log n)$. Each box has at most $2m(k-1) + d$ records, where d is the largest number of records with identical quasi-identifiers.
- Microaggregation: replace the records of each box by their centroid (Mondrian itself generalizes them to the box ranges).

Multivariate microaggregation example

Weight	Age	Pathogen
70	33	Virus
84	40	Bacteria
59	15	Virus
69	35	Protozoa
63	25	Virus
60	20	Bacteria

Microaggregation
 $k = 3$
applied to the
quasi-identifiers
Weight and Age

Weight	Age	Pathogen
74.3	36	Virus
74.3	36	Bacteria
60.7	20	Virus
74.3	36	Protozoa
60.7	20	Virus
60.7	20	Bacteria

- Each (Weight, Age) pair appears 3 times: 3-anonymity with respect to the quasi-identifiers.
- Pathogen is not modified. In the red cluster 2 of the 3 records have *Virus*: anyone known to be in that cluster has *Virus* with probability $2/3$ (attribute disclosure).

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

- Rank Swapping

- Microaggregation

- Noise**

- Generalization

- Other methods

- Synthetic Data Generators

Summary

Noise

- **Additive Noise:** add noise to the original data:

$$X' = X + r$$

- **Multiplicative Noise:** multiply the original data by noise:

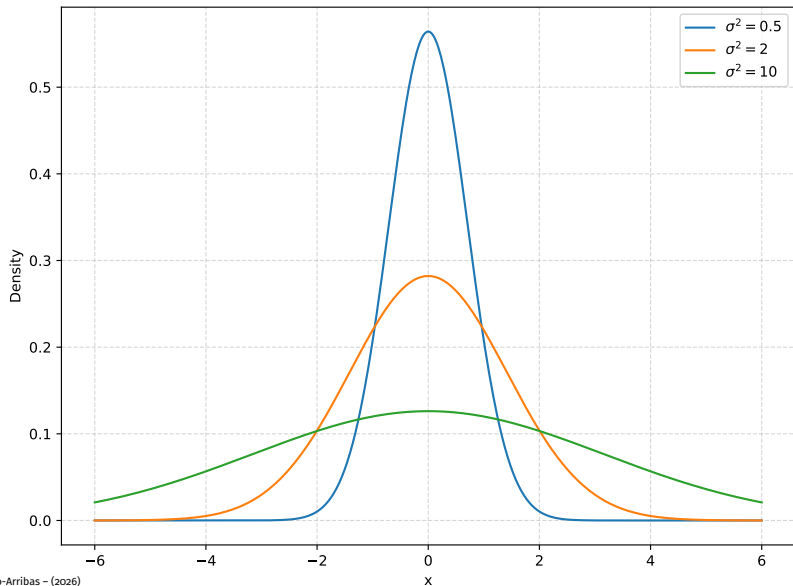
$$X' = X \cdot r$$

→ r follows a certain distribution (usually $E(r) = 0$ for additive noise, $E(r) = 1$ for multiplicative noise).

Additive noise

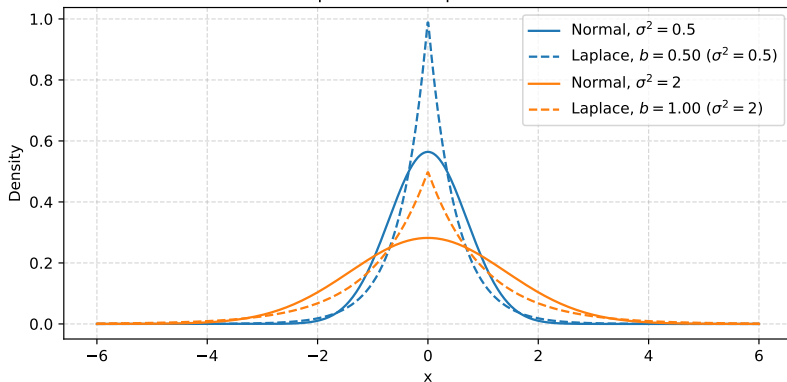
- Simple approach: add noise r_j to each attribute X_j , with $E(r_j) = 0$ and $\text{Var}(r_j) = p \text{Var}(X_j)$ (for a given constant p).
 - E.g. Normal distribution: $r_j \sim N(\mu = 0, \sigma^2 = p \text{Var}(X_j))$
- Bigger p means more distortion $\Rightarrow$ more privacy (and less utility).
- Noise from a Laplace distribution is commonly used to provide differential privacy (more on that later), but:
 - its scale depends on the sensitivity Δf of a query f and on ε ($b = \Delta f / \varepsilon$), not on $\text{Var}(X)$ [3];
 - it is usually added to query results (adding it to each record is the *local* DP setting).
- Noise proportional to $\text{Var}(X_j)$, as here, gives no formal guarantee.

(Recall) Normal distribution



Normal vs. Laplace noise

Same variance: Laplace is more peaked and has heavier tails



- Laplace($0, b$): density $\frac{1}{2b} e^{-|x|/b}$, variance $2b^2$.
- For the same variance: more small perturbations and a few larger ones.

Correlated vs. uncorrelated additive noise

Uncorrelated noise: $r_j \sim N(0, p \text{Var}(X_j))$, independently for each attribute.

- Preserves **means** and **covariances** between different attributes.
- Variances grow, $\text{Var}(X'_j) = (1 + p) \text{Var}(X_j) \Rightarrow$ correlations shrink: $\rho' = \rho / (1 + p)$.

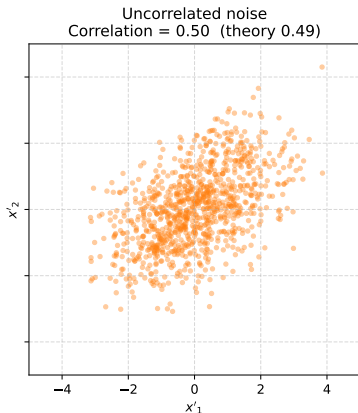
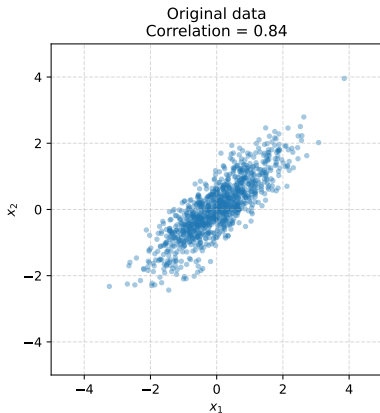
Correlated noise: $r \sim N(0, p \Sigma)$, where Σ is the covariance matrix of X .

- $\Sigma' = (1 + p) \Sigma$: preserves **means** and **correlation coefficients**; variances and covariances grow by $(1 + p)$.
- A linear transformation also restores the covariance matrix [9]:

$$X'' = \lambda X' + (1 - \lambda) \bar{X}, \quad \lambda = 1 / \sqrt{1 + p} \quad (\text{large } n).$$

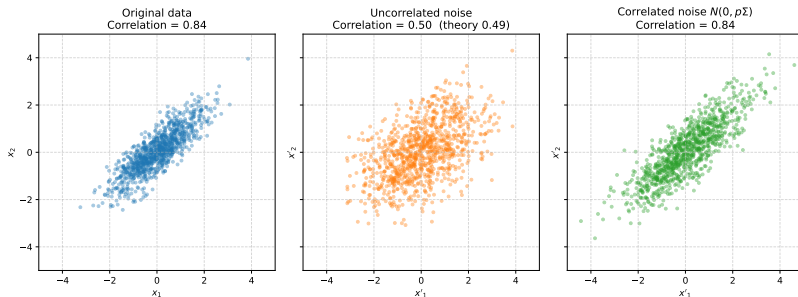
Example: noise and correlation

Uncorrelated noise with $p = 0.7$: $r_j \sim N(0, 0.7 \text{Var}(X_j))$.
Expected $\rho' = 0.84/1.7 \approx 0.49$.



Example: noise and correlation

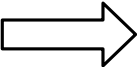
Same $p = 0.7$, uncorrelated vs. correlated noise
 $r \sim N(0, 0.7 \Sigma)$.



Additive noise example

Weight	Age	Pathogen
70	33	Virus
84	40	Bacteria
69	35	Protozoa
59	15	Virus
63	25	Virus
60	20	Bacteria

Additive noise example

Weight	Age	Pathogen		Weight	Age	Pathogen
70	33	Virus		71	36	Virus
84	40	Bacteria		81	40	Bacteria
69	35	Protozoa		68	30	Protozoa
59	15	Virus		58	16	Virus
63	25	Virus		62	29	Virus
60	20	Bacteria		59	23	Bacteria

- $r_j \sim N(0, p \text{Var}(X_j))$ added to Weight and Age (rounded to integers).

How much protection? Record linkage

- The attacker knows the quasi-identifiers (Weight, Age) of the individuals from an external source.
- Distance-based record linkage: link each protected record to the nearest original record. Re-identification rate = fraction of correct links.

Protection (previous examples)	Correct links	Re-id. rate
Additive noise	4 of 6	67%
Microaggregation, $k = 3$	1 per cluster of 3	33%

- Noise: only records (70, 33) and (69, 35) get confused. Small noise, little protection.
- Microaggregation: the 3 records of a cluster are identical, so the attacker can only guess. With k -anonymity the probability is at most $1/k$.

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

Rank Swapping

Microaggregation

Noise

Generalization

Other methods

Synthetic Data Generators

Summary

Generalization and Recoding

- Generalization of values following a given scheme or hierarchy of categories.

Age	ZIP	Profession	⇒	Age	ZIP	Profession
30	08193	Nurse		[30, 40)	081**	Health prof.
36	08176	Paramedic		[30, 40)	081**	Health prof.
32	08191	Veterinarian		[30, 40)	081**	Health prof.
25	08034	Mathematician		[20, 30)	080**	Science prof.
29	08022	Physicist		[20, 30)	080**	Science prof.
27	08010	Chemist		[20, 30)	080**	Science prof.

Global vs local recoding

- **Global recoding:** a value is always generalized in the same way, in all records.
- **Local recoding:** the same value can be generalized differently in different records.

Example: make (Age, City) 2-anonymous

Age	City	Global recoding	Local recoding
[20, 30)	Sabadell	Barcelona prov.	Sabadell
[20, 30)	Sabadell	Barcelona prov.	Sabadell
[30, 40)	Sabadell	Barcelona prov.	Barcelona prov.
[30, 40)	Terrassa	Barcelona prov.	Barcelona prov.
[30, 40)	Terrassa	Barcelona prov.	Barcelona prov.

- Global: consistent and easy to analyze, but more information loss.
- Local: less information loss, but the same attribute mixes granularities (Sabadell and Barcelona prov.).

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

- Rank Swapping

- Microaggregation

- Noise

- Generalization

- Other methods**

- Synthetic Data Generators

Summary

Other non-perturbative methods

- **Global suppression:** remove an attribute from the release.
- **Local suppression:** replace some values by a missing value to break rare combinations of quasi-identifiers.
 - E.g. $([30, 40), \text{Sabadell}) \rightarrow ([30, 40), *)$.
- **Top/bottom coding:** values above (below) a threshold are replaced by the threshold.
 - E.g. ages over 90 $\rightarrow$ "90+", incomes over 150k€ $\rightarrow$ " $\geq 150\text{k€}$ ".
 - Protects outliers, which are the easiest to re-identify.
- **Sampling:** release only a random sample of the records.
 - The attacker does not know if a given person is in X' .

PRAM: Post-Randomization Method

Perturbative method for **categorical** attributes [6].

- Markov matrix $P = (p_{ij})$, with
 $p_{ij} = \Pr(\text{released value} = j \mid \text{original value} = i)$.
- Each value is changed independently according to P .

P	Virus	Bacteria	Protozoa
Virus	0.8	0.1	0.1
Bacteria	0.1	0.8	0.1
Protozoa	0.1	0.1	0.8

- P is published: with frequencies T (original) and T' (released), $E(T') = P^\top T$, so $\hat{T} = (P^\top)^{-1} T'$.
- *Invariant* PRAM: choose P with $P^\top T = T$ (frequencies preserved in expectation).
- Closely related to randomized response (local differential privacy).

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

- Rank Swapping

- Microaggregation

- Noise

- Generalization

- Other methods

- Synthetic Data Generators**

Summary

Synthetic Data Generation

- Fit a model of the original data and sample new records from it.
- **Fully** synthetic (all records) or **partially** synthetic (only some attributes or values are replaced).
- No original record is released, but privacy is *not* automatic: the model can memorize and leak information about the training records (e.g. membership inference) [22].
- Approaches:
 - Statistical: CART (synthpop [15]), Bayesian networks, copulas.
 - Deep learning (currently very popular): GANs (e.g. CTGAN [25]), VAEs (TVAE), diffusion models.
 - Differentially private generators, with a formal guarantee: PrivBayes [26], MST [13].

GAN for synthetic data

- Generative Adversarial Networks [5] have been used to generate synthetic images and general data.
 - Generator G: produces synthetic data as similar as possible to the original.
 - Discriminator D: tries to distinguish between real and synthetic data.
- Different variations exist (e.g. CTGAN for tabular data).
- Caution: sometimes they are “too good” and can leak information from the original dataset.

Evaluating synthetic data

■ **Utility:**

- Compare marginals, correlations, ... (generic information loss).
- Train a model on synthetic data and test it on real data (specific information loss).

■ **Privacy:**

- Distance to closest record (DCR): are synthetic records (near) copies of original ones?
- Membership inference attacks: can an attacker tell whether a given record was used to train the generator?
- Only DP generators give a formal guarantee; the others have to be evaluated empirically.

Introduction to data privacy protection methods

Evaluation of protection methods

Protection methods

- Rank Swapping

- Microaggregation

- Noise

- Generalization

- Other methods

- Synthetic Data Generators

Summary

Summary of protection methods

Method	Type	Data	Preserves	Parameter
Rank swapping	Pert.	Num./ord.	Univariate distributions	p
Microaggregation	Pert.	Num. (mainly)	Means; k -anonymity	k
Uncorrelated noise	Pert.	Num.	Means, covariances	p
Correlated noise	Pert.	Num.	Means, correlations	p
PRAM	Pert.	Categorical	Frequencies (expected)	P
Generalization	Non-pert.	Cat./num.	Truthfulness	Hierarchy
Suppression	Non-pert.	Any	Truthfulness	–
Top/bottom coding	Non-pert.	Num./ord.	Truthfulness	Thresholds
Sampling	Non-pert.	Any	Truthfulness	Rate
Synthetic data	Synthetic	Any	What the model learns	Generator

Tools: statistical disclosure control and anonymization

- **sdcMicro** (R, GUI via sdcApp): microaggregation, noise, rank swapping, PRAM, recoding, suppression, risk measures [23].
<https://github.com/sdcTools/sdcMicro>
- **μ -ARGUS / τ -ARGUS**: SDC for microdata / tabular data, used by national statistical offices.
<https://github.com/sdcTools>
- **ARX** (Java, GUI): k -anonymity, ℓ -diversity, t -closeness, differential privacy, risk and utility analysis [18].
<https://arx.deidentifier.org>
- **pyCANON** (Python): checks which $k, \ell, t, \dots$ a dataset satisfies (k -anonymity, ℓ -diversity, t -closeness, ...) [20].
<https://github.com/IFCA-Advanced-Computing/pycanon>

Reference handbook: Hundepool et al. [8].

Tools: synthetic data

Generators:

- **synthpop** (R): CART and parametric models [15].
<https://CRAN.R-project.org/package=synthpop>
- **SDV** (Python): Gaussian copula, CTGAN, TVAE, ... [17].
<https://github.com/sdv-dev/SDV>
- **synthcity** (Python): many generators (incl. DP ones) and benchmarking [19]. <https://github.com/vanderschaarlab/synthcity>
- **MOSTLY AI Synthetic Data SDK** (Python): tabular and multi-table data, quality reports [10]. <https://github.com/mostly-ai/mostlyai>

Evaluation:

- **SDMetrics** (Python): utility (fidelity) and privacy metrics.
<https://github.com/sdv-dev/SDMetrics>
- **Anonymeter** (Python): privacy risk of synthetic data (singling out, linkability, inference) [4]. <https://github.com/statice/anonymeter>

References i

- [1] J. Domingo-Ferrer and J.M. Mateo-Sanz.
Practical data-oriented microaggregation for statistical disclosure control.
IEEE Transactions on Knowledge and Data Engineering, 14(1):189–201, January 2002.
- [2] George T. Duncan, Sallie A. Keller-McNulty, and S. Lynne Stokes.
Disclosure risk vs. data utility: The R-U confidentiality map.
Technical Report 121, National Institute of Statistical Sciences, 2001.
- [3] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith.
Calibrating noise to sensitivity in private data analysis.
In Theory of Cryptography Conference (TCC 2006), volume 3876 of LNCS, pages 265–284. Springer, 2006.
- [4] Matteo Giomi, Franziska Boenisch, Christoph Wehmeyer, and Borbála Tasnádi.
A unified framework for quantifying privacy risk in synthetic data.
Proceedings on Privacy Enhancing Technologies, 2023(2):312–328, 2023.
- [5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio.
Generative adversarial nets.
In Advances in Neural Information Processing Systems 27 (NeurIPS 2014), 2014.
- [6] J. M. Gouweleeuw, P. Kooiman, L. C. R. J. Willenborg, and P.-P. de Wolf.
Post randomisation for statistical disclosure control: Theory and implementation.
Journal of Official Statistics, 14(4):463–478, 1998.
- [7] Stephen Lee Hansen and Sumitra Mukherjee.
A polynomial algorithm for optimal univariate microaggregation.
IEEE Transactions on Knowledge and Data Engineering, 15(4):1043–1044, 2003.

References ii

- [8] Anco Hundepool, Josep Domingo-Ferrer, Luisa Franconi, Sarah Gießing, Eric Schulte Nordholt, Keith Spicer, and Peter-Paul de Wolf, editors.
Statistical Disclosure Control.
Wiley Series in Survey Methodology. Wiley, Chichester, UK, 2012.
- [9] Jay J. Kim.
A method for limiting disclosure in microdata based on random noise and transformation.
In *Proceedings of the Section on Survey Research Methods*, pages 303–308. American Statistical Association, 1986.
- [10] Ivona Krchova, Mariana Vargas Vieyra, Mario Scriminaci, and Andrey Sidorenko.
Democratizing tabular data access with an open-source synthetic-data SDK.
arXiv:2508.00718, 2025.
- [11] K. LeFevre, D.J. DeWitt, and R. Ramakrishnan.
Mondrian Multidimensional K-Anonymity.
In *22nd International Conference on Data Engineering (ICDE’06)*, pages 25–25, April 2006.
- [12] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam.
L-diversity: Privacy beyond k-anonymity.
ACM Transactions on Knowledge Discovery from Data, 1(1):3–es, March 2007.
- [13] Ryan McKenna, Gerome Miklau, and Daniel Sheldon.
Winning the NIST contest: A scalable and general approach to differentially private synthetic data.
Journal of Privacy and Confidentiality, 11(3), 2021.
- [14] Richard A. Moore.
Controlled data-swapping techniques for masking public use microdata sets.
Research Report Series RR96-04, U.S. Bureau of the Census, Statistical Research Division, 1996.

- [15] Beata Nowok, Gillian M. Raab, and Chris Dibben.
synthpop: Bespoke creation of synthetic data in R.
Journal of Statistical Software, 74(11):1–26, 2016.
- [16] Anna Oganian and Josep Domingo-Ferrer.
On the complexity of optimal microaggregation for statistical disclosure control.
Statistical Journal of the United Nations Economic Commission for Europe, 18(4):345–353, 2001.
- [17] Neha Patki, Roy Wedge, and Kalyan Veeramachaneni.
The synthetic data vault.
In *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 399–410, 2016.
- [18] Fabian Prasser, Johanna Eicher, Helmut Spengler, Raffael Bild, and Klaus A. Kuhn.
Flexible data anonymization using ARX—current status and challenges ahead.
Software: Practice and Experience, 50(7):1277–1304, 2020.
- [19] Zhaozhi Qian, Bogdan-Constantin Cebere, and Mihaela van der Schaar.
Synthcity: facilitating innovative use cases of synthetic data in different data modalities.
arXiv:2301.07573, 2023.
- [20] Judith Sáinz-Pardo Díaz and Álvaro López García.
A Python library to check the level of anonymity of a dataset.
Scientific Data, 9:785, 2022.
- [21] Agusti Solanas and Antoni Martínez-Ballesté.
V-MDAV: A multivariate microaggregation with variable group size.
In *Proceedings of the 17th Symposium on Computational Statistics (COMPSTAT 2006)*, 2006.

- [22] Theresa Stadler, Bristena Oprisanu, and Carmela Troncoso.
Synthetic data – anonymisation groundhog day.
In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1451–1468, 2022.
- [23] Matthias Templ, Alexander Kowarik, and Bernhard Meindl.
Statistical disclosure control for micro-data using the R package *sdcmicro*.
Journal of Statistical Software, 67(4):1–36, 2015.
- [24] Vicenç Torra.
Guide to Data Privacy: Models, Technologies, Solutions.
Undergraduate Topics in Computer Science. Springer International Publishing, Cham, 2022.
- [25] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni.
Modeling tabular data using conditional GAN.
In *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, 2019.
- [26] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao.
PrivBayes: Private data release via Bayesian networks.
ACM Transactions on Database Systems, 42(4):25:1–25:41, 2017.